Automated payment fraud detection using logistic regression and support vector machines

Heinrich Mathias Thétard

Thesis presented in partial fulfilment of the requirements for the degree of
Master of Commerce (Operations Research)
in the Faculty of Ecomnomic and Management Sciences at Stellenbosch University

Supervisor: Prof. JH Nel

March 2021

Declaration

By submitting this thesis electronically, I declare that the entirety of the work contained therein is my own, original work, that I am the sole author thereof (save to the extent explicitly otherwise stated), that reproduction and publication thereof by Stellenbosch University will not infringe any third party rights and that I have not previously in its entirety or in part submitted it for obtaining any qualification.

Date: March 2021

Abstract

The financial technology sector is a fast moving environment. There are many innovations in the automation and efficiency spheres where human intervention is required less and processing speed is rapidly increasing. In the payments space this is evident as payments are processed faster each year with the vast majority of these transactions driven automatically. This has opened up a platform for fraudsters to operate on.

The use of Machine Learning (ML) in fraud detection has grown in popularity. Two methods, logistic regression (LR) and support vector machines (SVMs), are used to identify fraud and are investigated in this thesis. LR is less complex as compared to SVMs, but SVMs have unique situations where it will outperform any other ML model [31]. Either method is assessed based on application conditions and measured based on a certain set of confusion matrix based metrics. The two methods are applied to a data set from a bank which participates in the automated payment environment.

It was evident that the sample proportions selected had a major impact on the model performance especially with regards to sensitivity and specificity. This was an exercise of fraud identification where sensitivity is the most important. This may not be the case for all data sets and environments as the cost to investigate false positives may be higher than the actual cost of fraud prevented.

Condition testing and post model application diagnostics were applied in this research. It was evident principle component analysis (PCA) feature selection was inferior to stepwise feature selection. The relatively poor performance of the PCA feature selection models is due to a loss of information when variables are removed when choosing the components.

When considering the odds ratios for LR, there were several variables that were protective factors and others that were risk factors. These factors either increased or decreased the odds of a case being fraudulent. It was found that when a debit order (DO) was associated with an older person it was more likely to be fraudulent than when the DO was associated with a younger person. It was also found that if a DO had a value of R99 or R45 then the odds of the case being fraudulent would increase several-fold.

LR models produced equivalent results to the more complex SVM models with a much better run time. From a practical point of view, this means that LR is preferred on larger data sets.

Key words:
Machine Learning; Logistic Regression; Support Vector Machine (SVM); Confusion Matrix; Debit Orders; Fraud Detection

Opsomming

Die finansiële tegnologie sektor is 'n vinnig bewegende omgewing. Daar is baie innovasies op die gebied van outomatisering en doeltreffendheid, waar menslike ingryping minder nodig is en die spoed van verwerking vinnig toeneem. In die betalingsruimte blyk dit dat betalings elke jaar vinniger verwerk word, met die oorgrote meerderheid van die betalingstransaksies wat outomaties verwerk word. Dit het 'n platform vir bedrieërs geskep. Gevolglik neem die gewildheid van die gebruik van masjienleer (ML) in die opsporing van bedrog steeds toe.

Twee metodes, logistieke regressie (LR) en ondersteuningsvektormasjiene (SVMs), word gebruik om bedrog te identifiseer en word in hierdie tesis ondersoek. LR is minder kompleks in vergelyking met SVMs, maar SVMs het unieke situasies waar dit beter sal presteer as enige ander ML-model. Elk van hierdie metodes word beoordeel op grond van toepassingsvoorwaardes en die prestasie word gemeet aan die hand van 'n sekere stel maatstawwe wat op die verwarringsmatriks gebaseer is. Die twee metodes word op 'n datastel van 'n bank wat aan die outomatiese betalingsomgewing deelneem, toegepas.

Dit was duidelik dat die geselekteerde steekproefverhoudings 'n groot invloed op die modelprestasie, sensitiwiteit en spesifisiteit gehad het. In hierdie studie is die identifikasie van bedrog die oogmerk, en daarom is die meting van sensitiwiteit die belangrikste. Dit is miskien nie die geval vir alle datastelle en omgewings nie, aangesien die koste om vals positiewe gevalle te ondersoek, hoër kan wees as wat die werklike koste van die voorkoming van bedrog is.

Die toetsing van voorwaardes en ontleding van postmodel diagnostieke is in hierdie navorsing toegepas. Dit was duidelik dat hoofkomponentanalise (PCA) ondergeskik presteer het in vergelyking met stapsgewyse seleksiemetodes. Die relatief swak prestasie van die PCA seleksiemodelle is te wyte aan die verlies van inligting wanneer veranderlikes ge-elimineer word in die keuse van die komponente.

By die oorweging van die kansverhoudings vir LR was daar verskillende veranderlikes wat beskermende faktore was en ander wat risikofaktore was. Hierdie faktore het die kans op gevalle van bedrog verhoog of verminder. Daar is gevind dat wanneer 'n debietorder (DO) met 'n ouer persoon geassosieer word, dit meer waarskynlik as bedrog geklassifiseer word as wanneer die DO met 'n jonger persoon geassosieer word. Dit is ook gevind dat as 'n DO 'n waarde van R99 en R45 het, die kans dat dit 'n bedrogsaak sal wees, meer sal vergroot.

LR-modelle lewer gelykstaande resultate aan die meer ingewikkelde SVM-modelle met 'n baie beter tydsduur. Uit 'n praktiese oogpunt beteken dit dat LR modelle verkies sal word vir groter datastelle.

Acknowledgements

Throughout the writing of this thesis I have received a great deal of assistance from many people including:

- Prof. JH Nel, who has always given me guidance, support and energy during the writing of this thesis. Thank you for the high standard that you have and know that I am proud to have had you as my thesis advisor.
- My parents, Rudi and Erika, who have supported my academic endeavours for the better part of two decades. This degree would not have been possible without your continued support and motivation. I thank you sincerely.

Table of Contents

List of Figures

List of Tables

List of Abbreviations

A/C	Authenticated Collection
AEDO	Authenticated Early Debit Order
AIC	Akaike Information Criterion
AUC	area under the curve
BIC	Bayesian Information Criterion
BIS	Bank for International Settlement
DD	Direct Debit
DO	debit order
EDO	Early Debit Order
EFT	Electronic Fund Transfer
KMO	Kaiser-Meyer-Olkin
KNN	k-nearest neighbours
LR	logistic regression
ML	Machine Learning
NB	naive Bayes
NAEDO	Non-Authenticated Early Debit Order
NPS	National Payment System
OLS	Ordinary Least Squares
PASA	Payments Association of South Africa
PCA	principle component analysis
PCH	Payment Clearing House
PSO	Payment Clearing House System Operator
RBF	Radial basis function
RF	random forest

ROC receiver operating characteristics

SAMOS South African Multiple Option Settlement

SARB South African Reserve Bank

SARS South African Revenue Services

SVM Support Vector Machine

VIF variance inflation factor

CHAPTER 1

Introduction

Contents

In a world of rapidly changing financial technology, security has become an ever pressing issue. The security of a financial system is critical to all participants of this system. Identifying security threats is an expensive and time consuming process, especially since many threats are only identified after they have caused damage. This means detection of threats like fraud have to be proactive. The need for evolving fraud detection methods is critical to the continued security of a financial system. The objective of this study is to describe what automated payment fraud looks like in South Africa and to present a possible solution for fraud identification using machine learning techniques. This chapter begins with the background of the debit order (DO) environment in South Africa, followed by the research questions and objectives. The research design, scope and expectations of the study are then discussed. The final component of this chapter is to outline the remaining chapters of the study.

1.1 Introduction and Background

The DO, internationally known as Direct Debit (DD), is a widely used payment method among banking clients in South Africa. The average monthly volume of Non-Authenticated Early Debit Order (NAEDO) transactions in quarter two of 2018 was 14.7 million debit orders; of which 1.9 million debit orders were disputed per month [43]. The volume of payments flowing through the debit order payment system is tremendous with the value of these debit orders equating to several trillion Rands per month.

A debit order is an instruction given by a client for a collector to collect funds, via an intermediary bank, from their account [53]. An example of this is a recurring payment [2], such as a utility bill from a client to a company, where the company initiates the payment collection. The client in this case has an account with Bank 1 and the company has an account with Bank 2. The

sponsoring bank is Bank 2 and the collector is the company. The company has a mandate to collect a certain amount of money from the client's bank account at Bank 1, where Bank 2 initiates the collection on behalf of the company on an agreed-upon date.

The interaction between Bank 1 and Bank 2 is managed by a Payment Clearing House (PCH) [68], or in South Africa it is known as a Payment Clearing House System Operator (PSO) [74]. The PSO will facilitate the payment and settlement between the banks.

The debit order function globally, and specifically in South Africa, has seen abundant use as a payment method due to the interoperability of banks. Interoperability is defined as: "The ease of interlinking different systems on a technological level" by the South African Reserve Bank (SARB) [44]. SARB goes further saying interoperable systems lead to the development of large network externalities, which decreases operational cost and increases the simplicity for the client due to economies of scale [44]. The degree of interoperability of the inter-bank payment systems is an indication of technological capability and demonstrates the maturity of a nation's financial sector.

There are multiple DO systems that operate within the debit order environment, namely the Electronic Fund Transfer (EFT) debit order and the Early Debit Order (EDO). Within the EDO environment there are two major subsystems: Authenticated Early Debit Order (AEDO) and NAEDO [4]. DebiCheck, which will be referred to as Authenticated Collection (A/C) for the remainder of the research, also falls into the EDO collection's environment with some notable differences to AEDO and NAEDO. These payment systems and their unique characteristics will be defined in greater detail in the next chapter.

The EFT is the most basic form of a debit order [74]. In such a transaction the mandate is granted to the collector by the client. In the case of an EFT, the amount deducted could be variable. A mandate is defined as: "an official order or commission to do something" [67], in the case of debit order collection the official order is given to collect money on behalf of a collector from a client's account. This is a contractual agreement in most cases and if the collection is rejected (or disputed) the contract is violated and legal recourse may be taken by the collector.

The EDO is a variant of the EFT debit order where collection is done earlier in the day [74]. The processing of these debit orders early in the day, specifically early in the morning (between 01H00 and 04H00), services a niche of debit order clients and collectors who require a debit order which runs immediately after the salary is deposited into the client's account [4].

The EDO is used widely in South Africa and is operated with or without an authorised mandate, giving rise to both the AEDO and the NAEDO. EDO was a significant change in the payment environment as it allowed collectors to pull money from accounts early in the morning before money could be withdrawn from an account. This decreased the risk of insufficient funds when collecting from a client's account and was particularly useful in the case of unsecured credit collections.

Clients are able to dispute DO payments and have the money refunded to their accounts. Most debit order disputes come about for two reasons [16]:

- the cash management DO dispute, where clients dispute the payment to a collector due to a lack of money and the need of the money. In this case legitimate debit order payments are disputed leading to abuse of the payment system [72];
- and the fraudulent DO dispute, where a client disputes a debit order payment because the debit order was fraudulent, after which a stop order is put into place to block future collection attempts.

Both types of debit order disputes have been cause for concern in the South African payment environment. The SARB considered the replacement of the NAEDO system. In 2013 the directive was given to develop the A/C payment system [54].

The existence of organised crime syndicates in South Africa has lead to large scale investigations into fraudulent debit order abuse amounting to at least R1.6bn per year being debited from South Africans [34]. Masela, head of the National Payment System (NPS), has said that A/C is designed to curb the fraudulent debit orders and remove the rogue elements from the NPS [34].

The abusive dispute behaviour has grown large enough to justify the implementation of an entirely new payment system. The classification of fraudulent and non-fraudulent DOs will be the main objective of this research.

1.2 Research Questions

The main research question is: Can fraudulent DOs be identified using machine learning techniques?

The supplementary research questions that will be answered are:

- what does the automated (DO) payment environment look like;
- what machine learning algorithms are available for this problem; and
- using a case study, can fraudulent DOs be identified reliably?

The research questions were used to define the research objectives.

1.3 Research Objectives

The main research objective will be to classify fraudulent and non-fraudulent DOs. A single, holistic case study of a bank that participates in the industry will feature the DO environment. Looking at the literature review and the case study separately, the below research objectives are listed.

A literature survey will be used to:

- describe the components of the payment system, interoperability and the four party model;

- define debit orders and the different types of payment systems;
- describe debit order disputes and abuse;
- describe fraud in the DO environment; and
- identify applicable models to identify fraud.

A case study will be used to:

- describe conditions for ML model application;
- apply ML models to the case study data; and
- define and apply model diagnostics and performance measurements to establish whether results are valid and reliable.

These objectives are linked to the research problems and aim to address the research questions. The research objectives are used to inform the research design.

1.4 Research Design

The ultimate goal for this research is to classify fraudulent DOs. Fraud in this area poses a risk to citizens and, if left unchecked, the problem will grow and become more difficult to contain. This is why it is imperative to identify fraudulent behaviour and flag DOs that look to defraud citizens.

For the above reasons a deductive research approach was chosen. With sufficient data and using machine learning techniques a model can be created that identifies fraudulent DOs. This research is exploratory in nature since the successful answering of the objective leads to insight gained in other areas. These areas are client behaviour and the mechanics of fraud in the automated payments space.

The use of primary data was not considered since high quality secondary data was obtained from a reputable bank within the automated payments environment.

1.5 Scope and Assumptions

The research will be restricted to EFT and EDO methods of payment, none of the other payment methods available in South Africa will be considered.

The data used will not be linked to any client in particular or in general.

In no instance will the details of clients be considered within the data. The source of the data (a bank and its partners) will not be identified. The data will always be aggregated to the highest possible level.

1.6 Expectations

There are several key expectations. The first expectation is to describe the current state of the debit order environment, by exploring the abuse within the automated payment systems and describing, in particular, fraudulent DO abuse.

Secondly, it is expected to classify DOs accurately (whether they are fraudulent or not) and to deliver a model that is built using field data as opposed to randomly generated data. It is expected that the results produced are valid and reliable.

1.7 Chapter Layout

This thesis will be broken into chapters roughly based on the layout specified by Bell *et al.* [5]. The remaining chapters are:

Chapter 2 and 3: Literature review

The necessary theoretical background is discussed in this section. The practical knowledge on DOs and ML are also discussed and analysed. The main objective is to understand the South African payments system as it applies to debit orders, how fraud occurs within the system and how machine learning techniques can be used to identify fraud.

Chapter 4: Research design and methodology

This chapter discusses the research design and methodology. Information about data collection, data analysis and presentation of results are provided in this section of the thesis. This will include the transformation of the data from the raw form to the final input into the classification model; how the models are validated and tested for reliability; and how the relative performance of each model is measured against one another.

Chapter 5: Results

In this section the results obtained in Chapter 4 are discussed. The data transformation, validity, reliability and model performance results are displayed and analysed. The models are compared and the necessary performance metrics are compared.

Chapter 6: Conclusions and recommendations

Using the results from the previous chapters, conclusions about the South African payments system and DO fraud are made. Conclusions and comparisons are made between the different models. The final recommendations and possible future studies are discussed.

This chapter summarised the basic DO environment and demonstrated the need for intervention as far as DO fraud detection goes. The research questions were stated followed by the research objectives. The research objectives will be reassessed in the final chapter. The basic research design was stated followed by the scope and expectations of this research. The final section gave the layout for the rest of the research. The first thing to consider is the literature regarding DOs and automated payments.

CHAPTER 2

Literature Review: Automated Payments

Contents

The purpose of this study is to identify fraudulent DOs using ML techniques. This chapter will describe fraudulent DOs which is the first major component of the overall objective. The first concept to be discussed is how a payment works and how it operates within the four party payment model. The interoperability between banks and clients is then discussed followed by a description of debit order payments. The two major groups of DO payments are discussed followed by a description of disputes and fraud in the DO environment.

2.1 Introduction

Globally many people use automated payments on a daily basis in one way or another. In fact, if an individual is paid a salary of a recurring amount, on a set date, it is considered an automated payment. Despite the vast use of automated payments, there is limited academic work done on DOs. There is more research available on DDs [36], the European equivalent to South Africa's DO.

Table 2.1 provides a list of keywords and databases used in the searches for this study. Finance, economics, government and legal databases were searched using specific keywords. The keywords searched on the different databases in Table 2.1 yielded several types of academically published material. This was predominantly limited to textbooks only accessible via universities, journal articles and government legislation gazettes.

Research Field	Database	Key words
Finance and Economics	EBSCO Host Scopus	Four party model Interoperability Debit order (or direct debit) EFT debit order Early debit order DO (or direct debit) disputes DO (or direct debit) fraud
Government and Legal	Sabinet Legal Westlaw (international law)	Four party model Interoperability Debit order (or direct debit) EFT debit order Early debit order DO (or direct debit) disputes DO (or direct debit) fraud

TABLE 2.1: *Table of keywords and databases searched.*

The concept of payments, the four party model, interoperability, and the broader idea of an automated payment (or debit order) are all widely researched in other countries where research has been conducted. For these sections there is a larger focus on academic material.

Sections which are unique to South Africa are: EFT DO, EDO, disputes in the debit order environment and fraud in the debit order environment. These payment systems and topics only occur in South Africa. This limits the scope of the academic literature on the above mentioned subjects to newspaper articles, government position papers and technical requirements of the system administrators.

The first concept to be discussed is a payment. It is a simple and well understood concept, but needs to be defined to give context to what a DO payment is.

2.2 Payments

A payment is a simple concept of the transfer of value (usually in the form of currency) from one party to another. The method in which the payment takes place can vary drastically, complicating the concept of a payment. The Bank for International Settlement (BIS) [13] claims that South Africa has a diverse payment environment, with highly sophisticated payment methods in the metropolitan areas. In rural areas a cash-based system is required. There is a need for multiple forms of payments in South Africa to service the needs of both sophisticated users and unsophisticated users.

The SARB is responsible for oversight and implementation of the NPS and has given the Payments Association of South Africa (PASA) the necessary jurisdiction to monitor the NPS [74]. The duties and responsibilities of the SARB include the provision of management, administrative, operational, regulatory and supervisory services for the NPS [52]. The SARB will

issue directives that contain binding rules addressing system risk and position papers containing guidelines to foster sound practice to PASA [13].

The PASA maintains the NPS which includes automated payments. The PASA assists the SARB with oversight of the NPS and the participating members by imposing penalties and fines for non-compliance. In 1998 the SARB introduced the South African Multiple Option Settlement (SAMOS) system which helped align South Africa with developed countries, allowing the South African NPS to be considered among the best in the world [13].

The banks of South Africa operate within the NPS and make use of systems such as SAMOS. The continued development and refinement of the NPS have allowed the financial sector, and therefore the banking sector, to become one of the most sophisticated in the world. Despite being considered one of the leading banking sectors globally, there is still fraud in the banking environment [64] and continuous action needs to be taken to eradicate the loopholes that exist in the system.

Having assessed the concept of a payment and the environment which regulates payments, it is necessary to describe how a payment works within this environment.

2.3 The Four Party System and Payment Clearing Houses

Before transactions were facilitated by banks, transactions used to be conducted via barter exchange or through a common, non-monetary currency. Banks came into existence and began facilitating transactions for clients who both belonged to the same bank, this system is known as the three party payment model [74]. In this case, there is a buyer, a seller and a bank.

The SARB makes use of a payment framework known as the four-party payment model [74]. A four party model has the following four parties in a transaction: a buyer, the buyer's bank, a seller and the seller's bank, refer to Figure 2.1.

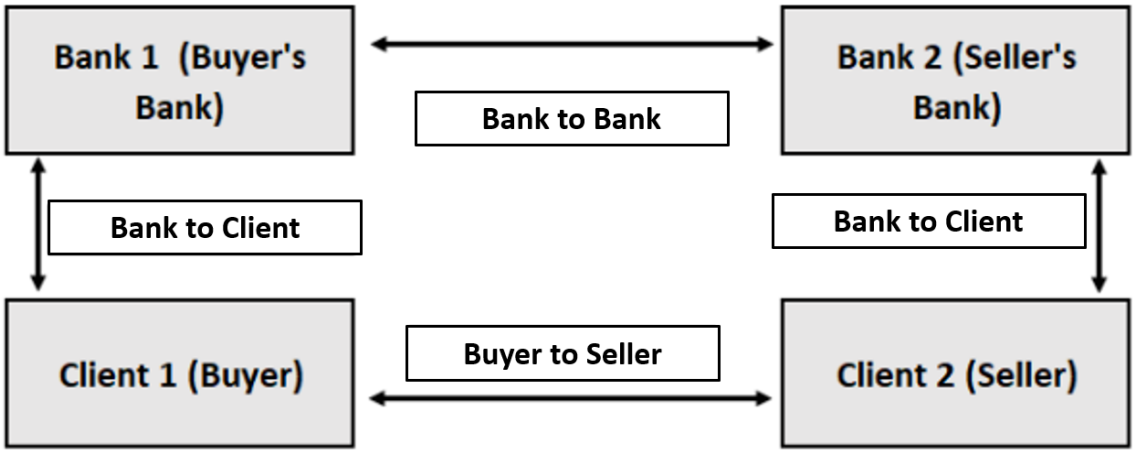

FIGURE 2.1: *The four party payment model.*

The SARB is responsible for the implementation and oversight of this payment system. The bank-to bank relationship is known as the interbank relationship and the SARB is the only institution allowed to settle interbank obligations in the form of cash or passing entries across individual books of banks [13].

According to Volker, there are three relationships that exist between the parties in the four party model [74]:

- buyer-to-seller: this relationship is commercial in nature. The rendering of a service or the sale of goods is always involved in this relationship;
- bank-to-client: this relationship is in the competitive sphere. Banks compete for clients by offering variable products in an attempt to increase their own market share and thus increase their profit; and
- bank-to-bank: this relationship is cooperative in nature, where increased cooperation allows for better service to be rendered to both banking clients.

The bank-to-bank relationship facilitates the execution of a debit order. Although the seller and the buyer are the main beneficiaries of the transaction, it is due to the presence and cooperation of the banks, in the interbank sphere, that the transaction is able to occur. The nature of a debit order is also that it is automated, meaning that the buyer and the seller do not need to intervene with the process once the debit order has been initiated.

This is where the concept of interoperability becomes important.

2.4 Interoperability

The need for interoperability in the NPS increased during the past 20 years. For each of the systems (EFT and EDO) banks needed to collaborate with each other, leading to an increase in electronic collaboration and ever increasing system complexity [23].

In many cases the word interoperability refers to the ability of computer systems to interact with each other. This is also the case for the NPS; however, the definition includes the extent to which institutions (PSO and banks) collaborate [44].

The benefits of achieving interoperability are:

- widespread access and convenience for clients as they can purchase using their bank cards at any seller and the payment will be processed correctly;
- increased customer value through increased functionality of their banking services;
- increased supplier competition and decreased monopolisation of certain industries; and
- positive network effects which give rise to economies of scale.

Economies of scale can be considered as the biggest benefit as it affects most people [74]. Volker [74] uses the telephone network example: if there are only two telephone users the cost-to-benefit of having the telephone is at its highest; however, the more people on the telephone network the lower the cost-to-benefit is. This means that the more people use the telephone network the more valuable the telephone becomes to each individual user.

The interbank environment is supplemented by an important institution known as a PCH, in South Africa the PCH is BankServ and is regulated by PASA. The PCH is responsible for settlement and clearance of transactions between banks [13]. Clearing is defined as: "the exchange of payment instructions" [68] by the SARB.

Debit orders are payments that move through the NPS and are a good example of implementing a system to the benefit of interoperability.

2.5 Debit Order Payments

A debit order is a type of automated payment. The ombudsman for banking defines the payment as: "an agreement between you and a third party in which you authorise or mandate the third party to collect money from your account for goods or services" [48]. Debit orders are frequently used for recurring payments such as mobile phone contracts or utility bills. In South Africa there are two types of debit orders in existence, EFT and EDO [74], with a third, A/C [51] to be fully introduced by the end of 2020. The introduction of A/C is directly as a result of problems with the EDO system. These problems are partly due to the placement of the mandate with the "Collector" in the four-party payment system, refer to Figure 2.2.

All debit orders require a mandate and the authorisation of the mandate. A mandate is defined as "an official order or commission to do something" [67]. In the case of debit orders the mandate is held by the seller (collector in Figure 2.2). This means that if a buyer (client in Figure 2.2) wishes to dispute the DO transaction, the money will be refunded [74]; however, the seller will have to produce the mandate as proof that the transaction is valid.

The fact that the mandate is held by the seller (collector) and not the banks or the PSO, creates the main loophole in the system that is abused by fraudsters. This is discussed in a later section of this chapter.

The main goal of debit orders is to increase convenience to the debit order user through automated payments of regular/recurring amounts and ease of interaction between banks for these payments. Important aspects of the debit order product are [2]:

- payment flexibility which allows for variable amounts to be debited;
- pre-authorised debit order collection; and
- a guarantee that funds will be refunded upon debit order dispute within a given time frame.

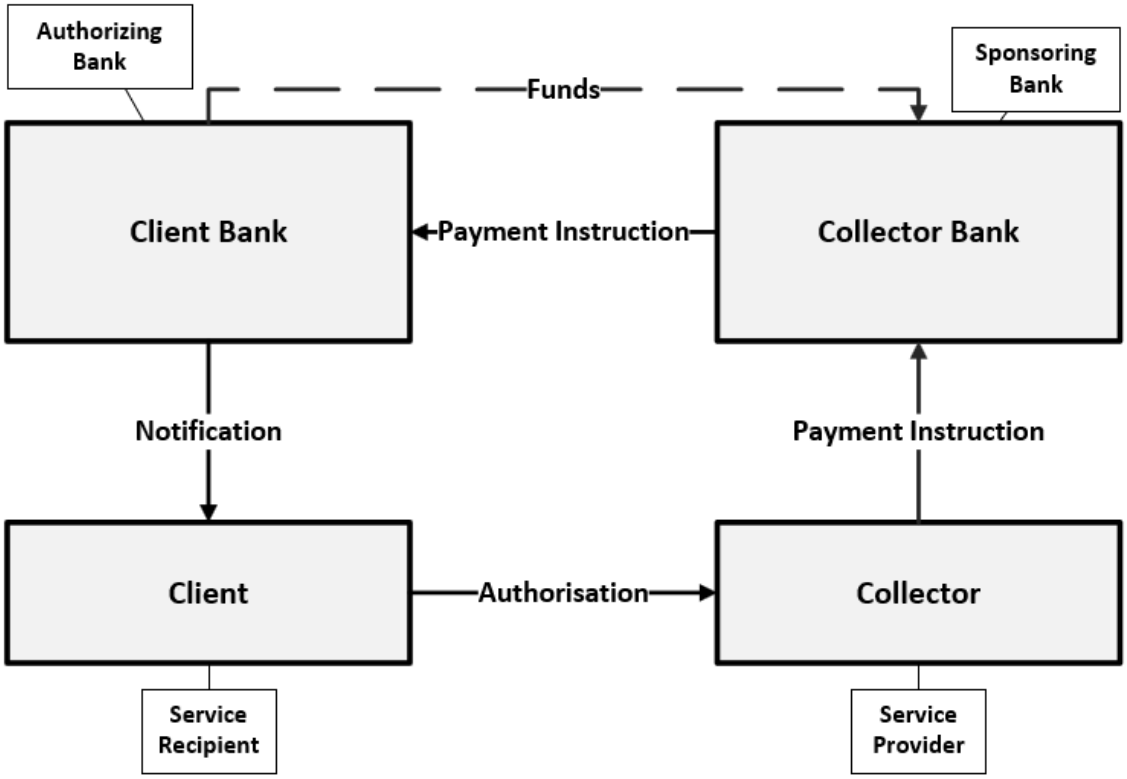

FIGURE 2.2: *Debit pull four party payment model.*

Debit orders may be disputed by the client in the following situations [48]:

- the client did not authorise the debit order that was debited;
- the deduction from the client's account is in contravention with the debit order's authority;
- the client instructed the collector to cancel the debit order; and
- the client stopped (or blocked) the debit order instruction.

The guarantee that cash will be refunded upon dispute of the debit order is a strong incentive for clients to dispute debit orders when they have limited funds [51]. This introduces one form of debit order dispute abuse known as "cash management" [72]. The second form of debit order disputes is fraudulent in nature and will be discussed later in this chapter.

The above section gives an overview of the concept of a DO payment in the context of South Africa. The following section will briefly describe the two largest DO systems.

2.6 EFT and EDO

The debit order that has been in use the longest is the EFT debit order. These debit orders are processed in batches at the end of each day [74]. The second category of debit orders are the EDOs and these are processed early in the morning [74]. There are two types of EDO debit

orders, namely AEDO and NAEDO [48]. The difference between the two is the authentication of the mandate. The AEDO is authenticated, meaning that the debit order is more difficult to dispute for the client. AEDO will not be discussed in detail in this review as it is largely a static service and is unaffected by fraud due to the authenticated mandate. The NAEDO is not authenticated which means that the burden of proof of the mandate falls on the collector if the debit order is disputed.

The EDO was introduced in 2006 to eliminate a problem known as "sorting-at-source", this is the process whereby the beneficiary of payment instructions sorts the payment instructions of each bank together and then submits those payment instructions directly to each paying bank [52]. The result of this was preferential treatment for certain parties; and the consequence was anti-competitive behaviour.

The EFT DOs and the EDOs allow for major movements in funds throughout South Africa. Abuse and fraud in these spaces lead to losses to both the general public and banking institutions. The next section will explore the idea of debit order abuse and how it is connected to fraud in the NPS.

2.7 Disputes and Fraud in the DO Environment

With the implementation of any new payment system there will be fraudsters and abusers that take advantage of loopholes in the system [14]. The EDO system, and more specifically the NAEDO system, is no exception. There are two types of abuse in the NAEDO system, namely cash-management disputes and fraudulent debit orders [51]. Both of these types of DO abuse will be described, beginning with fraudulent DOs.

The basis of many fraudulent debit orders (DDs) in Europe is identity theft, where a fraudster: "knowingly uses a means of identification of another person with the intent to commit, aid or abet any unlawful activity that constitutes a violation law" [22]. Coppolino *et al.* identify four different situations where fraud is committed using the debit order system, these four situations are [14]:

1. Location-independent service fraud: The goal of the fraudster is to benefit from a service without paying for it. Services included in this type of fraud are mobile phone contracts of media service providers;
2. Location-bound service fraud: The goal of the fraudster is to benefit from a service without paying for it. A gym membership is an example of a location-bound service fraud;
3. Address spoofing: The goal of the fraudster is to gain money. In this case the fraudster will steal an identity, set up a DD using this identity and order equipment to a location where the equipment can be collected by the fraudster; and
4. Fake company fraud: The goal of the fraudster is to gain money. The fraudster sets up a DD to debit an account without the knowledge of the account holder.

The most applicable situation in South Africa is fake company fraud. This is where a fraudster pretends to be a legitimate creditor, with the aim of setting up debit orders with a client and

collecting these debit orders under the pretence of being a legitimate company. Most of the debit order fraud in South Africa is committed using this method of defrauding clients [51].

A pensioner tells of her experience with fraudulent debit orders where four different companies debit amounts from her account that amount to R238 [56]. Since the amounts per individual debit order is below a certain threshold (usually R100), there is no message triggered to notify the victim that funds have been deducted from her account [9]. This way the debit order operates without the victim noticing. The pensioner mentioned earlier only noticed that she had these fraudulent debit orders when her legitimate debit orders failed to debit due to insufficient funds, after which she consulted her bank statement for more detail [56].

Certain individuals have had particularly negative experiences with debit orders, where they are defrauded for months, only realising after inspecting their bank statements that they had their money stolen. In addition to the loss of funds, the clients also have to pay fees for the fraudulent debit order. The final straw for many clients is when the banks facilitating the fraudulent debit orders say to the clients that "debit orders have nothing to do with them as it is between me (the client) and to whomever I give permission to take my money" [29].

Certain banks have taken action to combat the debit order abuse/fraud mentioned previously [9]. A certain South African bank traditionally had a debit notification limit of R100, where a prevalent scam would initiate debit orders at R99, maximising the amount debited and limiting the possibility of being uncovered. This scam is known as the "R99 scam". This bank decided to reduce the notification limit to R30 which meant that debit orders that operate at R99 would be identified, disputed and blocked [10].

Along with the increase in action taken by certain banks, there has been an increase in action taken by the regulatory institutions after the notable increase in disputes and fraud in the debit order space [69]. Clients were advised to check their bank statements on a regular basis to identify unauthorised transactions on their accounts [47]. This advice goes beyond debit orders. Consumers were also reminded that they have the right to dispute or instruct their bank to reverse debit orders they have not agreed to or are processed outside the mandate they have given [47].

The above behaviour in the market leads to an investigation conducted by the large banks and regulators in South Africa, SARB and South African Revenue Services (SARS). At the time the report was written three large debit order scam syndicates had been identified operating out of Kwa-Zulu Natal [10]. It has been estimated that as many as 750,000 fraudulent debit orders may have been implemented across the country's five major banks – totalling R74 million [10] per month.

The account numbers of clients are illegally obtained after which a company (fraudulent companies in most cases) then set up these payments without the knowledge of the client [47]. Once the account numbers have been obtained call centre operators contact the unsuspecting clients and then proceed to instate, with relative ease, debit orders, under a false company name, on the clients' account [64]. Despite efforts on behalf of the industry, blacklisting these fraudulent companies has been difficult since it only requires the name of the company to be replaced with a new name and the debit order will continue debiting the account [10].

The second major cause of disputes in the debit order environment is cash-management disputes. This is where a client disputes an already collected debit order in order to gain the refunded cash [51]. Clients dispute the legitimate debit order as the funds are reversed immediately and transferred back into the client's account which creates a temporary cash flow relief for the client [66]. This type of dispute falls outside the scope of this study.

Banks charge a fee for disputed debit orders which range in size. Debit order disputes lead to financial strain on consumers and it is particularly lower income groups that bear the brunt [45]. Since the client is in control of disputes the banks use their own discretion as to how large the penalty fee is. Certain banks charge an administration fee and other banks charge a punishment fee when a debit order is disputed which leads to further abuse of clients [3]. The unfortunate result of these larger penalty fees is that the client ends up in a worse financial situation [48].

Many clients suffer as a result of DO abuse, it is often the financially illiterate that suffer the most from these scammers. Banks are also using resources to curb the effect of DO abuse which could increase the quality of the client's experience.

The concept of a debit order and the fundamental principles of how a debit order functions were explored initially. Afterwards the idea of DO fraud was introduced with a description of how this problem comes about. The next chapter describes classification methods used to identify fraud.

CHAPTER 3

Literature Review: Machine Learning

Contents

The previous chapter shows how clients lose money to scams and how some banks combat the fraudsters. The use of machine learning presents the possibility to produce a more refined and long-term solution to the DO fraud problem. The literature pertaining to machine learning and payment fraud detection is explored further in this chapter.

The ML process will be described, followed by an in-depth discussion about LR. The LR section will explore the fundamental concepts surrounding the classification method as well as the conditions necessary for applying LR. The concept of feature selection will then be explored, followed by a discussion on model performance and confusion-based metrics. After the LR sections are discussed, SVM will be considered. The fundamentals of SVMs will be discussed first, followed by the conditions for applying SVMs. Feature selection and model performance applied to SVM will then be discussed. The final section of this chapter considers relevant fraud detection research.

3.1 Machine Learning

Identifying DO disputes is a major problem as it is abundant among all corners of society and is hidden among large amounts of raw data. Using ML methods to help in the identification of fraudulent DOs will allow individuals to identify and block fraudulent scams before they get a chance to infiltrate the individual bank accounts.

The purpose of ML is to learn from the data [15]. The machine learning algorithms can be grouped into three main categories based on knowledge about the output (Y) variable: supervised, semi-supervised and unsupervised learning [31]. Supervised learning requires a data set where the output (Y) variable is labelled, and the objective of modelling with this methodology is to attempt to classify the output variable as accurately as possible. Unsupervised learning on the other hand requires no labelled output variable, the objective of this modelling approach is to find patterns between variables and describe the structure within the data. Semi-supervised learning requires some of the observations with, and other observations without labelled output (Y) variables, with the aim of labelling unlabelled output variables [31].

The DO data used in the case study has enough labelled (as fraudulent or non-fraudulent) cases/observations to train a model with, so the main focus in this research will be supervised learning and the associated process, shown in Figure 3.1, is followed.

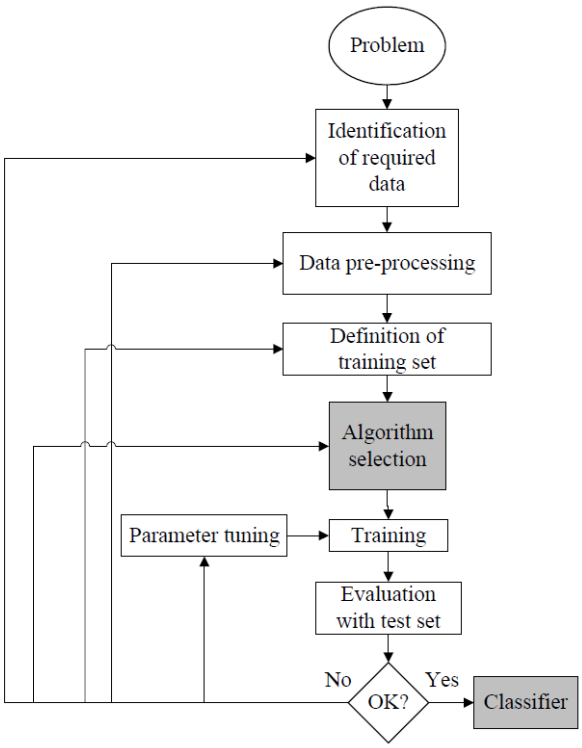

FIGURE 3.1: *Supervised machine learning workflow [37].*

Using the process flow shown in Figure 3.1, it should be possible to create a classifier capable of identifying fraud using the input variables (input variables will also be referred to as features) of the fraudulent (DO) records together with a classification method. The records are split into a training set and a test set. The model will be trained on the former, and tested on the latter; after which the model must be optimised to not over-fit the data and to reduce variance to a suitable level [31].

Since the DO data has a labelled output variable, the method used for classification will be in the supervised school of statistical learning [12]. The independent features will be modelled to classify the dependent categorical variables. Not all features are relevant to the categorical vari-

able so feature selection is performed to isolate only features that are relevant to the dependent variable.

Using the optimal subset of features produced from the feature selection process, a classifier needs to be tested on the data. Since the output variable, whether the DO is fraudulent or not, is categorical, a classification technique can be used for modelling [31]. There are two classifiers considered in the modelling process: LR and SVM. These methods are common in the fraud detection sphere as will be seen in Section 3.4. The reasons that these classification methods were chosen instead of other methods is because of the applicability to the dichotomous data set; the relative simplicity of LR compared to the complexity of SVM; both of these classification methods produced linearly separable decision boundaries; and they are often grouped together for comparison [31].

The model produced using an LR classifier has a good middle ground between the classification strength and the simplicity of interpreting the result produced [31]. The model produced by an SVM classifier does not have much interpretability, but has as high potential for good classification accuracy. SVM is frequently considered a "black box" classification solution since it gives a result, but is difficult to interpret. SVM is also considered an "out-of-the-box" solution since it produces good results without manual intervention [31]. Figure 3.2 below shows the trade-off between model flexibility and model interpretability. LR and SVM lie on opposite sides of this spectrum despite the fact that they are often compared to one another, as will be seen in Section 3.4.

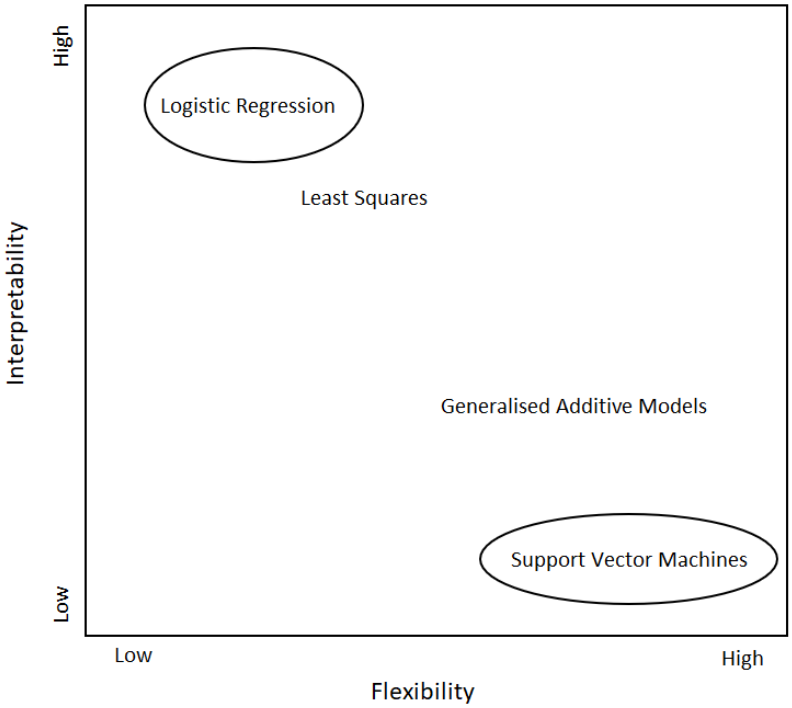

FIGURE 3.2: *Trade-off between model flexibility and interpretability [31].*

The first classification method to be discussed in detail is LR followed by SVM.

3.2 Logistic Regression

In this section the LR component of the modelling process will be discussed. The fundamental components of logistic regression are discussed initially, including concepts, conditions and reasons for application.

3.2.1 Fundamentals of Logistic Regression

LR is one of the most widely used classification methods and seeks to model the probability that an output variable belongs to a particular category or group [19]. The probability that the output variable belongs to a specific group always lies between 0 and 1.

The binary random (dichotomous) variable as defined by Dobson and Barnett [17] is:

$$Y = \begin{cases} 1 & \text{if the outcome is a success} \\ 0 & \text{if the outcome is a failure.} \end{cases} \tag{3.1}$$

The relationship between the input variable X and the binary response variable Y is given by equation (3.2). The functional form of the logistic regression function is given by

$$P = P(Y = 1|X = x) = \frac{e^{\beta_0+\beta_1 X}}{1 + e^{\beta_0+\beta_1 X}}, \tag{3.2}$$

where P is the probability that the binary output variable Y is a success, X is an input variable, β_0 is a constant and β_1 is the coefficient of input variable X.

With some manipulation the below equation can be produced,

$$(1 - P) = \frac{1}{1 + e^{\beta_0+\beta_1 X}}, \tag{3.3}$$

then the odds ratio is

$$\frac{P}{(1 - P)} = e^{\beta_0+\beta_1 X}. \tag{3.4}$$

The value $\frac{P}{(1-P)}$ is called the *odds*. This value can range from 0 to ∞ and indicates the likelihood of an event occurring. On average 1 in 5 DOs with an odds of $1/4$ are fraudulent, since $P = 0.2$ it implies an odds of $\frac{0.2}{(1-0.2)} = 1/4$, as an example [31].

Taking the log on both sides of equation (3.4) the result is

$$\ln\left(\frac{P}{1 - P}\right) = \beta_0 + \beta_1 X. \tag{3.5}$$

Equation (3.5) implies that β_1 has an impact on X. This impact is when X is increased by one unit the log odds changes by β_1, or equivalently it multiplies the odds by e^{β_1} [31].

The link between the linear function,

$$f(X) = \beta_0 + \beta_1 X, \tag{3.6}$$

and the "S" shaped sigmoid [31] function is given in equation (3.7). The link function [11] is

$$g(p_i) = \ln\left(\frac{p_i}{1-p_i}\right), \qquad i \in \{1, \ldots, n\}, \tag{3.7}$$

for a specific observation i, p_i is the probability of being classified as fraudulent, β_0 is a constant, β_1 is the coefficient of X and the total number of observations is n. Equation (3.7) is often referred to as the logit (or logistic) function for an observation y_i.

The shape of the logit function is shown in Figure 3.3. This solid "S" shaped line is when the logit of a binary variable, on the vertical axis, is plotted against a given input variable (X), on the horizontal axis. The probability of the binary variable being classified as fraudulent cannot exceed 1 and cannot be less than 0. For a given value of X an associated probability of Y being classified as fraudulent is given. For a cut-off value of 0.5, $p(Y) > 0.5$ gives a value of 1 and $p(Y) < 0.5$ gives a value of 0. The cut-off value used in this research is 0.5. This means that a continuous response variable $p(Y)$ is transformed into a dichotomous response variable.

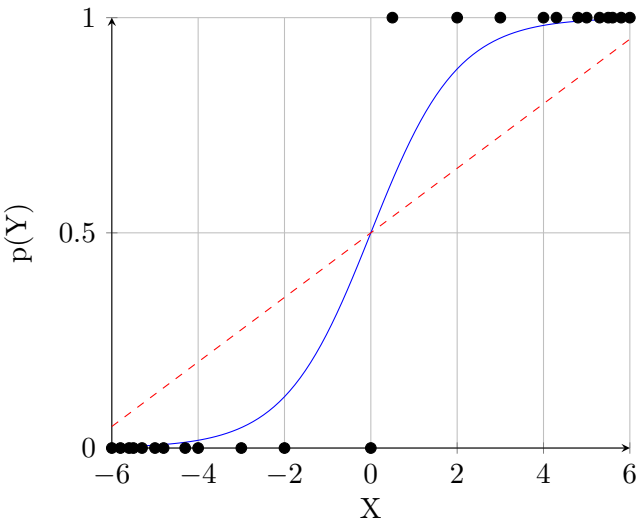

FIGURE 3.3: *Logit function for binary output variable.*

The dashed line in Figure 3.3 shows a linear equation $f(X) = \beta_0 + \beta_1 X$. Linear regression uses a straight line whereas LR uses a sigmoid shape to fit the distribution. The sigmoid shape fits the dichotomous output variable better than the linear model, as shown in Figure 3.3.

In many situations where an attempt is made to identify fraud, it is often required to flag fraudulent instances on a case-by-case basis. This means that the output variable is dichotomous

and logistic regression is an ideal model for such situations. This is seen in Table 3.5 where LR is applied in many studies. In addition to this ideal situation for applying LR there are additional advantages of using LR over other models. The main advantages of LR are:

- LR output can be analysed and the model is interpretable [31];
- LR makes no assumptions about the underlying distribution of the predictors [26]; and
- LR is relatively quick to train, specifically when compared to more complex models such as SVMs.

Having described the fundamental concepts and advantages of LR, it is necessary to describe the conditions for applying the model.

3.2.2 Conditions for Application of LR

The following list considers the conditions that need to be met before logistic regression can be applied to a data set:

- the dependent variable (Y) is dichotomous (binary) [35];
- the assumption of the linear relationship between the logit of the dependent variable and each continuous predictor variable (X) must be confirmed [26];
- multicollinearity must be removed [26]; and
- the sample size adequacy needs to be confirmed [26].

The above conditions need to be confirmed before the LR model can be applied. If the conditions are not confirmed then there is a possibility that the resulting model may, but not necessarily, have underlying issues. These issues may cause poor classification performance of the model.

Condition 1 (Dichotomy of output variable)

When fraud is considered it is necessary to assess each case individually for being fraudulent. This means that the fraud investigation has a binary output variable that can be either 1 or 0.

Logistic regression assumes an output variable which is dichotomous (binary) [35]. In the case of this research that assumption is met inherently as the dependent variable is whether a DO is fraudulent or not. The value of 1 is given to fraudulent observations and 0 to non-fraudulent observations.

Condition 2 (Independent variable linearity)

The LR classifier requires a linear relationship between continuous independent variables and the logit of the output variable [35]. This is inspected visually by plotting each continuous variable against the logit of the output variable.

The consequence of non-linearity in some of the variables is an inaccurate model. Since logistic regression is a linear model the decision boundary will be linear. Non-linearity in some variables will not fit the decision boundary and will decrease prediction accuracy.

Condition 3 (Multicollinearity)

When analysing data and building models around certain data sets a problem arises when some of the variables in the data set display similar behaviour to other variables in the data set. An example of such a situation is when one variable is used to create another variable, this is known as a derived variable. The derived variable may behave in the same way as the first variable, this is known as collinearity. Collinearity also exists between input variables when one input variable can be linearly predicted from the other input variables with a substantial degree of accuracy.

The presence of multicollinear variables in a data set means remedial action is required to remove the highly correlated variables from the data set before modelling takes place. The simplest way of detecting collinearity is by examining the correlation matrix of the input variables. When the absolute value of the correlation between two variables is high (close to 1) then collinearity is present [31]. Using this method with multicollinear variables is not that simple as a lower correlation value may be present for three or more variables that are highly correlated [31].

The degree of multicollinearity can be determined by calculating the variance inflation factor (VIF) scores. Equation (3.8) shows how the VIF is calculated. The VIF for variable j, V_j, is as follows [25]:

$$V_j = \frac{1}{(1 - R_j^2)}, \qquad j \in \{1, \ldots, k\}, \tag{3.8}$$

where R_j^2 is the coefficient of determination for the regression of X_j on the remaining k independent variables.

According to Mansfield and Helms [40] assessing VIFs is a good method for detecting multicollinearity. This method is superior to manual inspection of a correlation matrix since it is able to detect multicollinearity more objectively.

The VIF is used in the results component of this research. The smallest possible value for the VIF is 1, with values between 5 and 10 considered to be problematic [31]. Any variable with a VIF above 10 should be excluded. A VIF cut-off value of 5 will be used in this research.

According to James *et al.* [31], there are two approaches to dealing with collinear variables. The first is to combine the two variables into a single input variable and the second solution is to

simply drop one of the collinear variables. Dropping one of the two collinear variables does not hinder the modelling process as collinearity implies that the information contained in the one variable is contained within the other [31].

Condition 4 (Sample size adequacy)

Sample size adequacy is a required condition when applying LR classification [35]. Bujang *et al.* [8] recommend the following calculation for determining sample size adequacy:

$$n = 100 + 50k, \tag{3.9}$$

where n is the minimum sample size and k is the number of independent variables. The above calculation gives a recommended minimum sample size; however, in many real circumstances this minimum is very low and a sample size many times larger than the required minimum is used in the modelling process.

Each of the above conditions needs to be assessed when applying LR to a data set to ensure that the result is reliable. Once the conditions are assessed the next step in the ML process, feature selection, is considered.

3.2.3 Feature Selection and Extraction

A feature can be defined as: "an individual measurable property of the process being observed" [12]. To determine whether a DO is fraudulent or not, features (or variables) will be used to identify and classify the nature of the DO. There are many features associated with a DO and its user. Feature selection has three main advantages, namely: increasing the classification accuracy of the model by decreasing dimensionality; decreasing computational requirements; and increasing the general understanding of the data [12].

Reducing the number of features often reduces the variance of a model at a small expense of bias. This leads to significant increases in the model accuracy and the prediction accuracy on the test set [31]. Reducing the number of variables/features allows for greater interpretability as fewer variables need to be explained giving a less complex model as output [31]. Both of the above advantages lead to the third advantage which is decreasing the computational power required to produce the model. Since there are less input variables being fed into the model, the calculations required to produce the model decreases allowing for a quicker result.

Feature selection must not be confused with the removal of collinear variables. The presence of collinear variables may undermine the validity of a model; whereas, feature selection may increase the quality of a model. The removal of collinear variables is a necessity; however, feature selection can be thought of as a tuning factor for a model. Too many features/variables in a model may result in over-fitting and to few features may result in under-fitting [24].

Feature selection can be broken into three different methods for decreasing the number of variables. Firstly, subset selection seeks to reduce the number of input variables by removing those

that are believed not to influence the output variable. This is known as feature selection. Secondly, dimension reduction involves projecting the input variables into feature space and creating transformed variables in the process [31]. This is known as feature extraction. The third method is known as shrinkage. This method seeks to reduce the coefficient of a variable to as close to zero as possible. Ridge regression and the Lasso are two examples of shrinkage, but these methods will not be discussed in this research. Subset selection and dimension reduction will be discussed in the following sections.

Best subset selection (feature selection)

Subset selection algorithms can be classified into two main groups: filters and wrappers [42]. Filter methods seek to assess the relevance of each feature and wrappers sequentially/allegorically seek to develop a subset of features. Filter methods can often result in less than optimal subsets due to the inclusion of irrelevant variables [12]. Wrapper algorithms, such as forward, backward or stepwise selection, can be computationally intensive and do not always produce the optimal subset of variables either [12].

Wrapper algorithms, although not perfect, are still powerful and produce improved subsets of features. In many cases there is an increase in model performance using this feature selection method. The backward feature selection process is shown below [38]:

1. All of the features are fed to the regression model in the first iteration;
2. the p-value of each feature is assessed against a predefined threshold (usually 0.05);
3. the feature with the least significance, the largest p-value, is removed from the data set;
4. the reduced data set is fed back into the regression model and the next least significant feature is removed; and
5. the above process is repeated until the p-value of all features in the model are below the threshold.

Forward feature selection follows the same process; however, the data set begins with one feature. During each iteration of the regression model a new feature is added and assessed against a p-value threshold. If the feature has a significance above the threshold it is not considered as independent variable again.

The algorithms mentioned above can be combined to produce a stepwise feature selection [62]. This method has a higher chance of achieving the optimal set of features, but comes with the penalty of increased calculation time. Stepwise feature selection is known for being a "greedy" algorithm [38]. This is a general characteristic of the stepwise school of algorithms. The algorithm relies on trial and error to achieve an optimised subset of features at the cost of increased calculation time with no guarantee of increases in the model accuracy. The next section, PCA, approaches the concept of feature selection from a different point of view.

Forward, backward or stepwise algorithms use linear regression as the base learner and the stepwise feature selection method as the search procedure [38]. There are variations of the

procedure where the significance of a feature is not necessarily determined by a p-value, but rather by another performance metric such as model accuracy or Akaike Information Criterion (AIC).

Dimension reduction (feature extraction)

Dimension reduction methods, like PCA, use high dimensional space, also known as feature space, to reduce a large number of variables into a low-dimensional set of features. The result of PCA is an entirely new set of transformed features (or scores) that explains most of the variability in the data [31].

The use of PCA as a feature selection method requires two conditions to be met. Firstly, it needs to be determined whether the correlation matrix is appropriate; and secondly, the data needs to have a multivariate normal distribution [61].

Even though simply increasing the sample size improves the applicability of PCA, there is still a need to measure the degree of sampling adequacy. Sampling adequacy dictates whether a data set contains sufficient information to use PCA. The method for measuring sampling adequacy incorporates the Kaiser-Meyer-Olkin (KMO) statistic [32]. The KMO measure of sampling adequacy gives an indication of how well the data fits the PCA model by analysing whether a correlation matrix is appropriate for factor analysis. A KMO value below 0.5 is unacceptable [21]. According to Kaiser and Rice [33] values between 0.5-0.6 is "miserable" and values between 0.6-0.7 are "mediocre". A KMO cut-off value of 0.7 will be used in this research.

The KMO statistic K_j is calculated for a single independent variable j using the following equation [18]:

$$K_j = \frac{\sum_{i \neq j} r_{ij}^2}{\sum_{i \neq j} r_{ij}^2 + \sum_{i \neq j} p_{ij}}, \qquad j \in \{1, \ldots, k\}, \tag{3.10}$$

where r_{ij} is the correlation coefficient and p_{ij} is the partial covariance between variables i and j. For the overall KMO statistic K, the following equation is used:

$$K = \frac{\sum_{i \neq j} \sum_{i \neq j} r_{ij}^2}{\sum_{i \neq j} \sum r_{ij}^2 + \sum_{i \neq j} \sum p_{ij}}, \tag{3.11}$$

which considers all combinations of variables where $i \neq j$.

Normalisation is used to ensure that the independent variables come from a multivariate normal distribution [61], in some cases this is also referred to as standardisation or Z-score normalisation. This means that each feature is transformed to have the same scale, with a mean of 0 and a standard deviation of 1 (unit variance) [31]. The use of skewed data in PCA dimension reduction results in skewed component scores [63].

The variables are standardised using a scale transformation. The Z-score equation is as follows:

$$Z_i = \frac{X_i - \mu}{\sigma}, \qquad i \in \{1, \ldots, n\}, \tag{3.12}$$

where Z_i is the standardised score, X_i is the observation, μ is the average across the sample and σ is the standard deviation of the sample. The process of using PCA component scores as a feature selection method is described by James *et al.* [31].

Once feature selection is done the remaining features are used for classification modelling. At this point the model is applied and the resulting output is measured. The classification accuracy will be measured which constitutes the model performance.

3.2.4 Model Performance

Traditional methods such as AIC, Bayesian Information Criterion (BIC) or the coefficient of determination were used to calculate the performance of a model. These are proxy measures used when it was (and sometimes still is) not practical to evaluate the data using large data sets [31]. With improvements in computing and increased access to statistical software, the need to use these proxy performance measures was removed and model performance can be evaluated using test data sets. The use of a confusion matrix has now become a standard classification model performance measure, this will be seen in the final section of this chapter.

A confusion matrix used in conjunction with a classification model shows the predicted and actual classification [73] and allows for the measurement of performance of a model. The confusion matrix is used for two or more classes of the output variable. Since the DO classification problem has a binary classification the confusion matrix will be 2x2 matrix, with the layout as shown in Table 3.1.

		Actual	
		A - Positive	A^c- Negative
Predicted	B - Positive	TP	FP
	B^c- Negative	FN	TN

TABLE 3.1: *Confusion matrix layout.*

A positive outcome (the detection of fraud) gives a value of 1 and a negative outcome (no fraud detected) gives a value of 0. The actual values are variables from the data set and the predicted values are output variables from the classification model. The definitions of the matrix cells in Table 3.1 are as follows:

- TP - True Positive (predicted positive if the actual outcome is positive);
- FP - False Positive (Type I Error) (predicted positive if the actual outcome is negative);
- FN - False Negative (Type II Error) (predicted negative if the actual outcome is positive); and
- TN - True Negative (predicted negative if the actual outcome is negative).

The above definitions represent a sum of the number of observations that were classified to one of these four groups (TP, FP, FN and TN). Using the above mentioned classifications from the confusion matrix the performance measures, shown in Table 3.2, can be produced.

Measure	Formula
Accuracy [73]	$\frac{TP+TN}{(TP+FP+FN+TN)}$ (3.13)
Error [73]	$\frac{FP+FN}{(TP+FP+FN+TN)}$ (3.14)
Sensitivity (or Recall) [46]	$\frac{TP}{(TP+FN)}$ (3.15)
Specificity [31]	$\frac{TN}{(TN+FP)}$ (3.16)
Negative predictive value (NPV) [46]	$\frac{TN}{(TN+FN)}$ (3.17)
Precision (or PPV) [46]	$\frac{TP}{(TP+FP)}$ (3.18)
F1-measure [59]	$\frac{(2)(Recall)}{(Recall+Precision)}$ (3.19)

TABLE 3.2: *Confusion matrix metrics.*

The list of performance variables is extensive; however, the most important metrics, sensitivity, specificity and accuracy are discussed below.

Sensitivity measures the number of correctly classified fraudulent debit orders [46]. This value should be as high as possible. Sensitivity can also be referred to as the true positive rate.

Specificity as a measure of accuracy measures in the case of fraud correctly identified non-fraudulent cases. Specificity is used in the receiver operating characteristics (ROC) curve and plays a crucial role in the overall quality of a classification model.

Accuracy gives a general indication of how accurate the model classifies both true and false values [73]. In certain situations this measure is more important than sensitivity; however, in the fraud space, sensitivity is the more important measure. For all of the above measures the closer the result is to 1 (or 100%) the better the model is performing.

ROC

The ROC is a popular method of measuring the performance of a classification model. The graph combines both true positive (or sensitivity) and false positive (or 1-specificity) rates to show how accurate a model is [31].

Figure 3.4 shows an example of an ROC curve. True positive rates are considered on the vertical axis and false positive rates are considered on the horizontal axis. The dashed 45° line represents a classification model with a prediction accuracy of 50%, at 50% the model is effectively guessing each prediction. The solid curved line above the 45° line is the ROC curve.

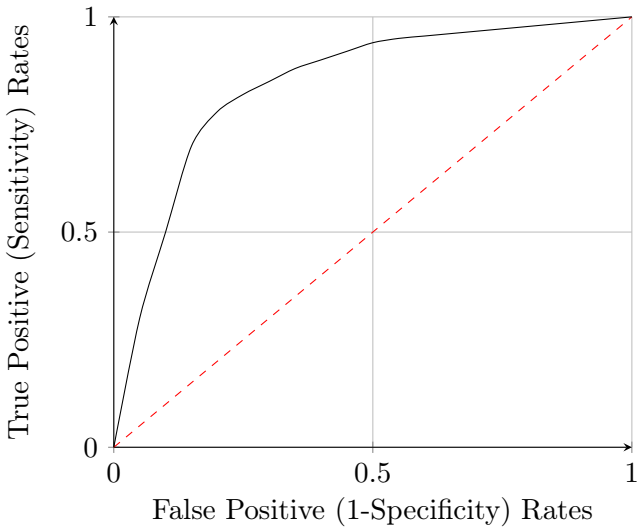

FIGURE 3.4: *ROC Curve.*

The better the classification performance of a model the higher the ROC will be above the 45° line. An ideal ROC curve will hug the top left corner [31]. The higher the ROC curve is above the 45° line the larger the area under the curve (AUC) is. The AUC should be high for a model to be considered to have high classification accuracy.

Both ROC and confusion matrix performance measures look specifically at classification accuracy; however, there are other performance measures to consider such as AIC.

AIC

The AIC is a measure that reflects the relative performance of a model and is often used as a method to choose between models. The AIC seeks to penalise complexity in models and reward simplicity, prioritising fewer independent variables over more independent variables [38].

The formula for AIC is [38]:

$$AIC = nlog(\sum_{i=1}^{n}(y_i + \hat{y}_i)^2) + 2k, \tag{3.20}$$

where n is the sample size, y_i is the output variable for observation i, $\hat{y}_i$ is the predicted value for the output variable for observation i and k is the number of input variables.

The AIC score of a model needs to be as small as possible and when models are compared using the AIC the model with the smallest AIC should be chosen. Considering equation (3.20), when all things remain the same, increasing k will increase the AIC value. AIC as a performance metric focuses on simplicity and the way to achieve the simplicity is to limit the number of features to include only the essential features.

McFadden's Pseudo Coefficient of Determination

McFadden's Pseudo R^2 is designed to measure the predictive capability of a model [39]. The pseudo R^2 measure, like many other measures, needs to be used in conjunction with other performance metrics. McFadden suggested a suitable minimum value for when a model fits the data well. The value should be between 0.2 and 0.4 [39].

McFadden's Psuedo R^2 considers the predictive capability of a model and also serves as a diagnostic check for models.

Hosmer-Lemeshow

The Hosmer-Lemeshow test is used to determine the goodness-of-fit of an LR model on a given set of data [26]. The test indicates how well the model fits and is used as a diagnostic test. The predicted values are arranged from the lowest to the highest and the cases are then divided into groups with approximately equal sizes, treating the known fraudulent and non-fraudulent cases separately. The observed and expected frequencies are compared in each group within the known fraudulent and non-fraudulent groups separately. The recommended number of groupings is 10; however, this number can vary depending on the data set [57].

The test produces a Chi-squared value along with a p-value. The significance level for the p-value is 0.05. The hypothesis test associated with the Hosmer-Lemeshow test is shown in Table 3.3.

Rejection of the H_0 hypothesis does not always imply that there are issues with the model. Using Hosmer-Lemeshow and McFadden's Pseudo R^2 together gives a clearer view of the quality of

H_0 :	The LR model fits the data.
H_α :	The LR model does not fit the data.

TABLE 3.3: *Hosmer-Lemeshow hypothesis test.*

the model. An example would be to reject the H_0 hypothesis of the Hosmer-Lemeshow test and to obtain an acceptable McFadden Pseudo R^2 score. This would mean the model predicts sufficiently well, but there might be underlying issues present in the model which compromises the goodness-of-fit.

Although the Hosmer-Lemeshow test is popular there are issues with the test. Hosmer specifies the following two shortcomings [27]: the test does not take over-fitting into account, and the choice of the number of groups is determined by the user which can influence the resulting statistic.

Paul, Pennell and Lemeshow [50] suggest that an increasing sample size can have an undesired effect on goodness-of-fit tests since small departures from the proposed model are considered significant when the sample size is large. The following equation suggests the number of groups required per sample size:

$$g = 2 + 8\left(\frac{n}{1,000}\right)^2, \tag{3.21}$$

where g is the number of groups and n is the sample size. Equation (3.21) implies that for n = 500, g = 10 groups are required; for n = 4,000, g = 130 groups are required; and for n = 2,5000, g = 5,002 groups are required. It is suggested that the Hosmer-Lemeshow test cannot be applied to sample sizes above n = 25,000 since it is required that there should be at least 5 observations per group to prevent an overpowered test [50].

The above-mentioned issues lead to the power of the Hosmer-Lemeshow test often being considered as low and it is recommended to use the statistic in conjunction with other diagnostic statistics.

Calculation time

Another method, used in conjunction with the above-mentioned measures, is to measure the calculation time of each of the models. Some models are mathematically more complex and this results in higher computation times. The larger a data set grows the higher the computation time becomes as well. For this reason the computation time will also be included in the results and used as a comparitive metric.

The above sections described the fundamental concepts underpinning the LR model, followed by conditions required to apply the model. The concept of feature selection was discussed, after which the model performance metrics and diagnostics were discussed. The above section focused specifically on LR; however, the feature selection methods and model performance measures can also be applied to the SVM models.

3.3 Support Vector Machine

Having looked at LR, the next section focuses on SVM as a classification method. SVM is a more complex classification method than LR; however, the two methods have many components that overlap and, as will be seen in Table 3.5, they are often considered in conjunction with one another.

This section of the chapter starts with a description of the fundamentals of the SVM classification method, followed by conditions required to apply SVM and methods for measuring the classification performance.

3.3.1 Fundamentals of Support Vector Machines

The support vector machine is a classification method which can be used on binary output variables. It was developed in the computer science community and can be compared to other classification methods such as logistic regression [31] since it was originally designed for binary classification [41].

The SVM is a progressed version of an "intuitive classifier" called the maximal margin classifier [31]. The objective of this classifier is increasing the separation distance between two groups of observations as much as possible. The separation of the two groups is done via an optimal separating hyper-plane which is trained by observations that lie on the edge of either group [41], these are known as the support vectors [6].

Equation (3.22) is the expression for a hyperplane, where β_0 is a constant, $\beta_1, \ldots, \beta_p$ are coefficients and $X_1, \ldots, X_p$ are observations:

$$\beta_0 + \beta_2 X_1 + \beta_2 X_1 + ... + \beta_p X_p = 0. \tag{3.22}$$

Note that in two dimensions, (3.22) is the equation of a straight line [31].

The maximal marginal classifier builds on the concept of a hyperplane by attempting to separate two groups of observations as far as possible. The maximal margin hyperplane is constructed based on n training observations $x_1, \ldots, x_n \in R^p$ with the associated class labels $y_1, \ldots, y_n \in \{-1, 1\}$ [31]. The maximal margin hyperplane is the solution to the following optimisation problem [31]:

$$\underset{\beta_0, \beta_1, \ldots, \beta_p}{\text{maximise}} \quad M \tag{3.23}$$

subject to

$$\sum_{j=1}^{p} \beta_j^2 = 1, \tag{3.24}$$

$$y_i(\beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + ... + \beta_p x_{ip}) \geq M \qquad \forall \ \ i = 1, \ldots, n, \tag{3.25}$$

where M is the width of the margin. M should be made as large as possible for an optimal solution. The maximal margin classifier cannot be applied to most data sets since it requires that the classes be separable by a linear boundary with no overlapping observation from the other group. So another method needs to be used when groups overlap, this is where the support vector classifier can be used.

The support vector classifier is an extension of the maximal margin classifier which is the solution to the following optimisation problem [31]:

$$\underset{\beta_0, \beta_1, \ldots, \beta_p, \epsilon_1, \ldots, \epsilon_n}{\text{maximise}} \quad M \tag{3.26}$$

subject to

$$\sum_{j=1}^{P} \beta_j^2 = 1, \tag{3.27}$$

$$y_i(\beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + ... + \beta_p x_{ip}) \geq M(1 - \epsilon_i), \tag{3.28}$$

$$\epsilon_i \geq 0, \tag{3.29}$$

$$\sum_{j=1}^{P} \epsilon_i \leq C, \tag{3.30}$$

where M is the width of the margin, C is a non-negative tuning parameter and ϵ_i is the slack variable.

The solution to the above optimisation problem involves the inner products of the observations and not the observations themselves [31]. The inner product of the two observations $x_i, x_{i'}$ is given by

$$\langle x_i, x_{i'} \rangle = \sum_{j=1}^{P} x_{ij}, x_{i'j}. \tag{3.31}$$

The general form of the linear support vector classifier can be shown in the following form

$$f(x) = \beta_0 + \sum_{i \in S} \alpha_i \langle x, x_i \rangle, \tag{3.32}$$

where α_i are the coefficients of feature vectors x_i and x_j, $i = 1, \ldots, n$, one per training observation and S is the sample [31]. Replacing the inner product in the equation (3.32) with a generalisation of the inner product form

$$K(x_i, x_{i'}), \tag{3.33}$$

results in

$$f(x) = \beta_0 + \sum_{i \in S} \alpha_i K(x_i, x_j), \tag{3.34}$$

where α_i are the coefficients of feature vectors x_i and x_j, $i = 1, \ldots, n$, one per training observation and S is the sample [31]. The kernel function is denoted by $K(x_i, x_j)$ in equation (3.34).

The support vector classifier solves the problem of overlapping groups, but it still produces only a linear solution [31]. In many cases the groups overlap but are not separable by a linear classification boundary, this is where the support vector machine becomes very relevant. The support vector machine function is shown in equation (3.34) with the general kernel function being $K(x, x_i)$.

Using kernels the SVM transforms the data by projecting the observations into a hypothetical space known as a feature space [31]. Many different kernels exist for use by SVMs. Alternative kernels allow for "varying degrees of non-linearity and flexibility" which may increase the applicability of the model [19]. Possible SVM kernels can be seen in Table 3.4.

The observations are mapped into the feature space using the kernel, once the observations are in a higher dimensional space, then the SVM seeks to create a separating hyper-plane between groups of observations. The hyper-plane can then be used as a function to classify unlabeled observations [71]. Thus, the SVM can find a linear solution to a non-linear problem, as shown in Figure 3.5.

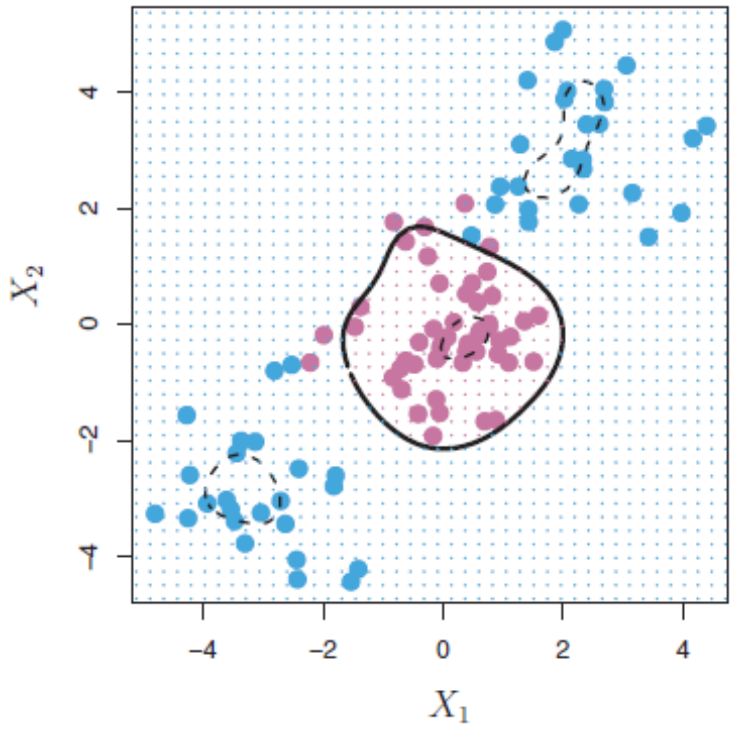

FIGURE 3.5: *Non-linear boundary produced by a hyperplane in feature space [31].*

It is difficult to find reasons why specific kernels are better than others. It seems researchers choose a few kernels and see how they perform based on a certain output measure. In many cases of research, priori knowledge of the data and environment are used to select the kernel [71]. Each kernel and SVM combination will have tuning parameters, often algorithms are used to optimise the parameters of the model. Ben-Hur and Weston [6] point out that the specific choice of kernel is "data-dependent", and recommends testing several kernels (and parameters) to try and find the optimal kernel.

Referring to Table 3.4, according to Hsu *et al.* [28] the RBF kernel is a reasonable first choice. This is because the RBF kernel is better at modelling non-linearity than the linear kernel and in certain cases the RBF kernel is equivalent to both the linear and sigmoid kernels [28]. When the RBF kernel is compared to the polynomial kernel, the RBF kernel is much less complex since the polynomial kernel has more hyper-parameters [28].

When the number of features is very large the application of an RBF kernel may not be suitable and the use of a linear kernel will be more appropriate [28]. One of the advantages of SVM, discussed in more detail later, is its applicability to small data sets or data sets where the number of features are more than the number of observations (n observations $<$ p features).

The most popular kernels are shown in Table 3.4; however, this is not an exhaustive list of kernels.

Kernel function	Formula	Parameter
Linear kernel [65]	$K(x_i, x_j) = x_i.x_j$ (3.35)	
Polynomial kernel [6]	$K(x_i, x_j) = (x_i.x_j + 1)^d$ (3.36)	d
Radial basis function (RBF) kernel [6]	$K(x_i, x_j) = exp(-\gamma\|\|x_i.x_j\|\|^2)$ (3.37)	$\gamma < 0$
Sigmoid kernel [6]	$K(x_i, x_j) = tanh(b(x_i.x_j) + c)$ (3.38)	b,c

TABLE 3.4: *Table of kernels [65] [6].*

The applicability of SVMs to data sets with many features and few observations is one of several advantages of the classification methods. The following are additional advantages of the modelling method:

- when compared to more widely used classification methods like decision trees and neural networks, SVMs may be more accurate [41];

- SVMs works well on small samples, producing high accuracy on small data sets [41];
- from a financial point of view, the high accuracy from small data sets produced by SVMs means a financial saving in data acquisition [41]; and
- the SVM model can easily move from a linear classifier to a non-linear classifier.

The following are disadvantages of SVMs:

- the output of the SVM is purely dichotomous with no class membership probability [19];
- the results of SVMs are difficult to interpret due to the fact that the method uses a "black box" and only gives a binary result [31]; and
- the SVM has a complex mathematical setup that results in increased computational power required to perform the calculations as the sample size increases. This is costly from a time consumption perspective.

It is interesting to note how the relatively high calculation time of SVMs on larger data sets is a disadvantage, but the relatively high classification accuracy on smaller data sets demonstrates the duality of the advantages of SVMs.

Having looked at the fundamental concepts surrounding the SVM it is necessary to consider the conditions required for apply the model.

3.3.2 Conditions for Application of SVMs

Compared to LR the SVM has fewer conditions. There are two conditions that need to be met before SVMs are applied.

The first condition for the use of SVMs is normalisation of the input data. According to Ben-Hur and Weston [6] certain situations of SVM classification can be highly sensitive to the way data is scaled. It is highly recommended to normalise the data before applying the classification model. This will become part of the conditions required for the application of SVMs as a classifier in the research methodology component. Equation (3.12) from the LR section can be used to standardise the data.

The second condition that is required for use of SVMs is the removal of multicollinearity [38]. This means the same VIF diagnostics (using equation (3.8)) performed when applying LR can be performed when applying SVMs.

Once the data is prepared and all conditions are met the feature selection process can be applied to the data set.

3.3.3 Feature Selection and Model Performance

Mention was made earlier of the similarities between LR and SVM. This is due to the fact that many of the diagnostic and validation methods used for LR can be used for SVM.

Non-informative variables can negatively affect SVMs [38]. Two feature selection methods are specified above in the LR section of this chapter. Both of these feature selection methods will be used to remove non-informative variables from the data before applying SVM.

The model performance for the SVM is done in the same way as with LR. Confusion matrix measures (see Table 3.2) are used to assess the performance of SVMs. ROC curves will be considered for each permutation of the model along with calculation time for the training of the model.

The fundamental concepts of LR and SVMs are discussed in the sections above. The conditions for applying each method and feature selection methods are also discussed. The confusion matrix performance measures are discussed for both methods as well. The following section will consider the above theory and will discuss how other researchers have approached the problem of identifying fraud.

3.4 Relevant Fraud Detection Research

The use of ML to identify fraud is not a new concept and is widely used in the credit card industry. In this section we explore the application of ML to detect fraud and critically assess what other researchers have found in their research. Table 3.5 shows a summary of relevant research in the field followed by a discussion of the major findings.

Itoo *et al.* [30] consider the use of machine learning methods on credit card fraud detection. The methods suggested are logistic regression, Naive Bayes and KNN. Itoo *et al.* [30] indicate the problem of skewed data where there is a much larger proportion of non-fraudulent cases than fraudulent cases, random under-sampling is used as an attempt to overcome this. Random under-sampling is explained as deliberately taking proportionally less observations from a selected group, comparing to another group. In this case non-fraudulent cases are under-sampled to increase the proportion of fraudulent cases in the test sample. The confusion matrix based performance metrics are used to assess the relative performance of each classification method. It is found that logistic regression performs the best on the data used in the research, also having the highest sensitivity.

Peussa *et al.* [55] suggest the use of logistic regression based score cards for credit default detection. It is postulated that a "gut feel" approach was used traditionally to determine the ability of a credit applicant to repay credit. Traditionally the overarching idea of the credit granting methodology was non-probabilistic. Peussa *et al.* [55] suggest the use of probability based score cards as a replacement for the traditional method, with specific use of LR to determine the credit default rate. AIC and BIC are used to measure the performance of the classification method. It is found that the statistical (LR) score card method is superior to the non-statistical "gut-feel"

Researcher	ML Methods	Year	Industry	Evaluation Method	Relevance
Itoo *et al.* [30]	LR, naive Bayes (NB) and k-nearest neighbours (KNN)	2020	Credit card	Confusion matrix methods	Yes
Peussa *et al.* [55]	LR	2016	Credit score card	AIC and BIC	Yes
Yu and Wang [76]	Distance Sum Method	2009	Credit card	Confusion matrix methods	No
Awoyemi *et al.* [1]	KNN, LR and Naive Bayes	2017	Credit card	Confusion matrix methods	Yes
Singh *et al.* [65]	SVM	2012	Credit card	Confusion matrix methods	Yes
Oza [49]	LR and SVM	2018	Credit card	Confusion matrix methods	Yes
Bhattacharyya *et al.* [7]	LR, SVM and random forest (RF)	2011	Credit card	Confusion matrix methods	Yes
Subudhi and Panigrahi [70]	SVM	2015	Tele-communications fraud	Confusion matrix methods	Yes
Ravisankar *et al.* [60]	SVM, LR, genetic programming and neural networks.	2011	Financial statement fraud	Confusion matrix methods	Yes

TABLE 3.5: *Summary of research done on fraud detection.*

method.

Awoyemi *et al.* [1] suggest the use of ML for credit card fraud detection. The data is a real-world set of credit card transactions. KNN, Naive Bayes and LR are suggested as possible methods for classification. Two major problems with fraudulent data are explored: the first is the unbalanced (skewed) nature of fraud data and the second is the continuously changing client spending behaviour in the credit card market. Proportional sampling is suggested as a method to deal with the unbalanced data. Confusion matrix based methods are used to assess the performance of the model. LR performed with low accuracy and the other two methods performed with high accuracy. There is no evidence of diagnostic testing post modelling, there may be an underlying issue with the data causing the LR method to perform relatively poorly.

Singh *et al.* [65] propose the use of SVM as a credit card fraud detection method. It is indicated that the customer spending profile is the traditional measure for detecting fraudulent credit card transactions; the use of SVM is suggested as an alternative to this. Singh *et al.* [65] suggest the use of multiple kernels, including the linear, quadratic and RBF kernel. The confusion matrix based performance metrics are used to measure the performance of each model. It is found that the RBF kernel outperforms all other kernels.

Oza [49] proposes the use of LR and SVM in credit card payment fraud detection. A data set with labeled observations is used, dictating the use of supervised machine learning methods. Oza [49] indicates the problem of false positives and the effect it has on the modelling process. Proportional sampling is not used; however, weighted classes (fraudulent and non-fraudulent) are applied in the classification process to overcome the imbalanced data problem. Confusion matrix based performance measures are used in this research. A comparison is made between LR and SVM and it is found that both methods perform with high levels of accuracy and minimal false positives (high sensitivity rate). It is also stated that: "in a payments fraud detection system, it is more critical to catch potential fraud transactions than to ensure all non-fraud transactions are executed smoothly" [49].

Bhattacharyya *et al.* [7] assessed the feasibility of LR, SVM and RF for credit card fraud detection. In this study real-life data is used as opposed to simulated data. The models aim to classify a fraudulent credit card transaction out of many non-fraudulent transactions, the primary output variable is dichotomous. The concept of feature selection is explored in an attempt to increase the performance of each model; however, detail regarding the selection methods used is limited. Confusion matrix based performance metrics are considered for assessing the relative performance of these models. The classification methods used show sufficient capability to be used for fraud detection, despite the fact that the data is imbalanced.

Subudhi and Panigrahi [70] investigate the feasibility of using SVMs, in different forms, to identify fraudulent call patterns in a telecommunications network. Anomalies are detected when the current call pattern does not match the historical call pattern of a given user. A real-world dataset is used when applying SVM where relative success is seen. Confusion matrix based performance metrics are considered for assessing the relative performance of these models. Subudhi and Panigrahi [70] believe that this method of anomaly detection can be extended to other spheres as they were able to classify fraudulent instances with success.

Ravisankar *et al.* [60] suggest the use of ML for financial statement fraud detection. Techniques considered are SVM, LR group method of data handling, genetic programming and two derivative methods of neural networks. Feature selection is explored in the research and features are selected based on their ranked t-statistic. Confusion matrix based performance metrics are used. Probabilistic neural network performs the best when no feature selection is applied to the data set; genetic programming and probabilistic neural network performed the best when feature selection is applied. Ravisankar *et al.* [60] suggest that "feature selection is critical to data mining".

Several authors ([7], [49], [1], [76], [30]) indicate the presence of unbalanced data in the case of fraud. This is due to the large proportion of non-fraudulent cases compared to the relatively small proportion of fraudulent cases. This means that when a sample is drawn from the population of fraudulent data, there is a small proportion of fraudulent observations compared to an overwhelming number of non-fraudulent observations. It has been suggested that standard machine learning algorithms will not perform well for imbalanced data sets since the algorithms were built with the assumption of a balanced distribution [75]. Traditional classifiers that aim to maximize the overall prediction accuracy tend to classify all data into the majority class [20].

Proportional sampling is suggested to overcome this problem. Itoo *et al.* [30] find that there is a trade-off between sensitivity and specificity based on the sample proportion chosen. It is

found that a high sensitivity may lead to a low specificity, this is due to a large number of non-fraudulent cases being classified as fraudulent cases, leading to a high false positive rate. Random under-sampling (or over-sampling) is another method which has been found to improvement in the trade-off between sensitivity and specificity [75]. Several authors ([30], [75], [20]) suggest a sample proportion that is not representative of the population class proportions. This research will explore the use of proportional sampling to overcome the issue of unbalanced data.

There is a problem with the false positive rate of the models due to the unbalanced nature of fraudulent data. The ability of a model to obtain a high fraud detection rate (true positive rate) often comes at the expense of a high false positive rate. Both LR and SVM are capable of dealing with this issue to a certain degree; however, the main objective is to identify fraudulent cases, so sensitivity (true positive rate) is of critical importance. Oza [49] makes a comparison between LR and SVM and it is found that both methods perform with "high levels of accuracy and minimal false positives".

Peussa *et al.* [55] suggest the use of AIC and BIC as performance measures for their classification model. This is postulated to be a traditional method of classification performance measurement. James *et al.* [31] suggest that this method was initially used due to the limitations of computational power. Applying complicated models to large data sets with limited computation power is not practical so proxy measures, such as AIC and BIC, were used. Since this is no longer the case, and computing power allows classification models to be applied to ever larger data sets, the use of confusion matrix performance metrics have become the predominant method for classification performance measurement. Confusion matrix performance measures are used by several authors ([30], [1], [65], [49], [7], [70], [60]).

There were some interesting choices of performance metrics in some of the articles, such as the use of AIC and BIC by Peussa *et al.* [55]. Post modelling diagnostics were not mentioned and it is unclear whether it was not done or simply not mentioned. Awoyemi *et al.* [1] found that LR fit the data quite poorly; however, there was no post implementation diagnostic evidence to support this.

In the majority of the above research there was no mention of feature selection. Whether this means that it simply was not mentioned or not done is unclear. Bhattacharyya *et al.* [7] explore feature selection to a certain extent, but Ravisankar *et al.* [60] suggest that "feature selection is critical to data mining". Feature selection is explored in more detail in this research.

There is a large focus on binary classification methods, such as LR and SVM. This is due to the ability of the model to classify dichotomous output variables successfully. Oza [49] compared these classification methods directly and found that they perform relatively well in the credit card fraud space. This research will focus on LR and SVM as classification methods and will be explored in more detail in the following chapter.

Most of the research on fraud detection using ML methods came from the credit card fraud detection space. This is not the same as the DO (or automated payment) space; however, many of the principles, such as the four party payment model, are the same. It is logical to extend the knowledge from the credit card fraud sphere to the DO fraud sphere. Subudhi and Panigrahi [70] believe that this method of anomaly detection can be extended to other spheres.

There are many important learnings from the related fraud research. There is a common issue with unbalanced data which can be addressed by proportional sampling; however, there is a trade-off between sensitivity and specificity depending on the sample proportion. Two very applicable methods in the fraud classification sphere are LR and SVM, both of which were discussed in greater detail earlier in the chapter and were commonly used by other researchers in this field. The idea of feature selection was explored in the technical literature review, but was seldomly used by other researchers. Confusion matrix performance metrics was discussed earlier in the chapter and were used extensively by other researchers in the fraud detection field. There was limited mention of pre- and post-diagnostic tests on models in the literature review; however, this was discussed earlier in the chapter and will form part of the platform of reliability in the results section.

All of the above mentioned topics will form part of the modelling process which will be discussed in the methodology chapter which follows this chapter.

CHAPTER 4

Research Methodology

Contents

In this chapter, the process used when applying LR and SVMs to the real world data is described. First, data preparation is described, followed by a description of the conditions necessary for applying LR and SVMs. The feature selection process is then described, followed by the methodology for applying LR and SVM. The final section to be discussed is model validation which entails post application diagnostics and model performance measures.

4.1 Introduction

A single case study of one of the participating banks operating in the debit order environment is performed. The overarching idea is to conduct a holistic case study in an attempt to model the data to be able to identify fraudulent DOs using machine learning techniques, specifically using LR and SVMs as classifiers.

In this chapter, a detailed description of the research methodology is given; however, the summary steps of the quantitative analysis and machine learning application are:

1. data (quality) preparation (process applicable to both LR and SVM):
   (a) data preparation and extraction;
   (b) multicollinearity assessment using VIF (using equation (3.8));
   (c) conditions for LR;

- step 1: multicollinear variables removed in step 1(b);
- step 2: sample size needs to be assessed (using equation (3.9));
- step 3: linearity of continuous independent variables is visually inspected;

(d) conditions for SVM with RBF kernel;

- step 1: normalisation to achieve unit variance and remove the effect of feature scale (using equation (3.12));
- step 2: multicollinear variables removed in step 1(b);

(e) feature selection applied to the cleaned data set:

i. stepwise feature selection using p-value significance; the process is specified in Chapter 3;

ii. PCA feature selection, beginning with the condition testing;

- step 1: KMO statistic used to determine sampling adequacy (using equation (3.10));
- step 2: standardisation used to achieve multivariate normality, (using equation (3.12));
- step 3: the PCA feature selection process is then applied;

(f) splitting the data into test and training sets;

2. model application:

(a) apply the LR;

(b) apply the SVM with the RBF kernel;

3. model validation based on the below metrics (process applicable to both LR and SVM):

(a) ROC curves to obtain the AUC values;

(b) Hosmer-Lemeshow test for goodness-of-fit for LR (hypothesis test in Table 3.3);

(c) McFadden's test for predictive power for LR;

(d) AIC measure for relative comparison of model fit (using equation (3.20));

(e) accuracy, sensitivity and specificity (consult Table 3.2 for equations); and

(f) calculation time.

The steps shown above will now be described in more detail beginning with the data preparation phase.

4.2 Data Preparation

The purpose of the data preparation phase is to extract and transform the data for use in the classification model. The focus is on preparing the data used by the model, so certain conditions need to be met for the classification model to be valid.

The data used in the modelling process was gathered by a bank participating in the NPS. Secondary data is defined as: "data gathered and recorded by someone else prior to the current needs of the researcher" [77]. The DO data is secondary data and was obtained from a willing

banking institution. Written permission was given by the institution and ethical clearance was obtained from the university forum.

The DO data is given on a transaction level and has demographic information per transaction, ultimately representing a portion of the automated payments that exist within South Africa. Data collected during the year of 2019 is considered for analysis, and include several million observations. A maximum limit of five million observations is used for the modelling component. This limit is set due to the increasing computation time as the number of observations increase. The full list of variables can be found in the Table 4.1 below.

TABLE 4.1: *List of variables from DO data set..*

Begin of Table 4.1			
Number	Variable Name	Variable Definition	Data Type
X1	Action Date	Calendar date that the DO occurred on.	Date
X2	Age Band	Age of DO client.	Ordinal
X3	Amount	Value of debit order (per DO).	Continuous
X4	App Activated	DO clients who currently have an activated app.	Binary
X5	App Registered	DO clients who are currently registered for the app.	Binary
X6	ATM Withdrawal Amount	Value of withdrawals a DO client has made from an ATM, DNR, External ATM or Spark ATM during the month.	Continuous
X7	ATM Withdrawal Amount Grouped	Grouped value of withdrawals a DO client has made from an ATM, DNR, External ATM or Spark ATM during the month.	Ordinal
X8	Banking Client	1 if DO client qualifies as a banking client, 0 otherwise.	Binary
X9	Branch Name	Sign up branch name.	Categorical
X10	Branch Visits 12 Month	A proxy of the number of times the DO client has visited a branch in the past 12 months.	Ordinal
X11	Branch Visits 12 Month Grouped	A proxy of the number of times the DO client has visited a branch in the past 12 months.	Ordinal
X12	Branch Visits 6 Month	A proxy of the number of times the DO client has visited a branch in past 6 months.	Ordianl
X13	Branch Visits 6 Month Grouped	Grouped number of times the DO client has visited a branch in past 6 months.	Ordinal
X14	Business Manager	Business manager code.	Categorical
X15	Bank Employee Employee	Current-, past- and non-bank employee.	Categorical
X16	Client Key	DO client key.	Categorical
X17	Credit Card Client	1 if DO client has a credit card, 0 otherwise.	Binary
X18	Current NLR Score	The NLR (National Loans Register) score of the DO client in the relevant month.	Continuous
X19	Current CPA Score	The CPA (Credit Card) score of the DO client in the relevant month.	Continuous
X20	DO Current Month	Number of debit orders in the current month.	Ordinal
X21	DO Current Month Grouped	Grouped number of debit orders in the current month.	Ordinal
X22	Employer Key	The most recent employer key details captured for the DO client.	Categorical
X23	Fixed Savings Client	1 if DO client is a fixed savings client, 0 otherwise.	Binary

Continuation of Table 4.1

Number	Variable Name	Variable Definition	Data Type
X24	Flexible Savings Client	1 if DO client is a flexible savings client, 0 otherwise.	Binary
X25	Gender Code	Gender of DO client.	Binary
X26	Government	Identifies a client who is employed by the government.	Binary
X27	Grouping	DO outcome (disputed, successful or insufficient funds).	Categorical
X28	Inflow Current Month	Total currency inflow into DO client account.	Continuous
X29	Inflow Current Month Grouped	Grouped total currency inflow into DO client account.	Ordinal
X30	Month end Date	Month of analysis.	Date
X31	Most Frequent Branch Key	Branch key which was visited most by DO client.	Categorical
X32	Most Frequented Branch Name	Branch name which was visited most by DO client.	Categorical
X33	Most Recent Visit Branch Key	Most recent visit branch key.	Categorical
X34	Most Recent Visit Branch Name	Most recent visit branch name.	Categorical
X35	Num ATM Withdrawals Current Month	Number of withdrawals a DO client has made from an ATM, DNR, External ATM or Spark ATM during the month.	Ordinal
X36	Num ATM Withdrawals Current Month Grouped	Grouped number of withdrawals a DO client has made from an ATM, DNR, External ATM or Spark ATM during the month.	Ordinal
X37	Official Client Type	Official DO client definition. This is a sub-segmentation of all active DO clients. This is either: loan, fee, save, new.	Categorical
X38	Operations Manager	Operations Manager code.	Categorical
X39	POS Current Month	The average number of pure POS (card swipes) per DO client per month.	Ordinal
X40	POS Current Month Grouped	The grouped value of pure POS (card swipes) transactions.	Ordinal
X41	POS Value	The value of pure POS (card swipes) transactions.	Continuous
X42	POS Value Grouped	Grouped value of pure POS (card swipes) transactions.	Ordinal
X43	Province	Province of the DO client's most frequented branch.	Categorical
X44	Quality Banking Client	1 if DO client qualifies as a quality banking client, 0 otherwise.	Binary
X45	Quality Banking Client Desc	Grouping if DO client is either non-banking, banking or quality banking client.	Categorical
X46	Quality Reverter	DO Client reverts from a quality banking client to a banking client.	Binary
X47	R45Flag	1 if DO value is R45, 0 otherwise.	Binary
X48	R99Flag	1 if DO value is R99, 0 otherwise.	Binary
X49	Reference	System reference of DO (per DO).	Categorical
X50	Reference Clean	Abbreviated name of DO initiator.	Categorical
X51	Regional Manager	Regional manager code.	Categorical
X52	Reverter	DO Client reverts from a banking client to a non-banking client.	Binary
X53	Risk Group NLR Score	The NLR (National Loans Register) Score of the DO client in the relevant month.	Ordinal
X54	Stable Inflows	Currency inflow to accout for three or more months.	Categorical

Continuation of Table 4.1

Number	Variable Name	Variable Definition	Data Type
X55	Stable Inflows Code	Currency inflow to accout for three or more months code.	Categorical
X56	Stable Product Usage	Consistent product usage for 3 or more months.	Categorical
X57	Stable Product Usage Code	Consistent product usage for 3 or more months code.	Categorical
X58	Term Loan Client Good Standing	No arrears on loans.	Binary
X59	Val Group	Grouped value of DO.	Ordinal
X60	USSD Registered	Book view of total DO clients registered for USSD.	Binary
X61	USSD User Current Month	Has used USSD for at least 1 transaction during the month.	Binary
X62	App User Current Month	Has used the app for at least 1 transaction during the month.	Binary
X63	DO Current Month	The number of debit orders deducted from the DO client's account per month.	Binary
Y	Group Classification	0 if non-fraudulent DO, 1 if fraudulent and 0 if DO status is unknown. Only DO status of 1 and 0 is considered in this research.	Binary

End of Table 4.1

Table 4.1 gives the number of the variable in the first column. The variable name is given in the second column, the variable definition is given in the third column and the fourth column contains a description of the type of variable.

In Table 4.1, the only continuous variables are where the term *Amount* can be found in the variable name, such as *X6*. The remaining variables are categorical or ordinal variables, including binary variables.

The final variable Y is the output variable and represents whether the observation is fraudulent, non-fraudulent or unknown. The modelling is focused on predicting this variable.

There is a mixture of financial indicators, demographic indicators and behavioural indicators. The data used is real-life data and has not been simulated. It can be seen that there are derived variables in the list which will be removed in subsequent steps of the methodology. Examples of the derived metrics are variables *X54* and *X55*, which are both indicators of account inflow stability.

Several of the variables, such as *X43* (Province), are non-ordinal categorical variables. These variables will be transformed into dummy variables. In the case of variable *X43* the "Head Office" province classification will be made the reference category since the largest amount of DO fraud exists in this province, "Head Office" is additional to the branches in the 9 provinces. These dummy variables will be coded as new variables and the result of these alterations will be shown in Chapter 5.

4.2.1 Data Preparation and Extraction Steps

The data extraction from the server was performed using Transact-SQL and R [58]. The first step is to change all text groupings into integer groupings as logistic regression requires integer groupings. This step was done in the Transact-SQL phase. An example is when the variable *X26* (Government) has either a government or non-government status, this was converted to 1 for government status and 0 for non-government status. The code for this example can be seen in Figure 4.1.

```
,case
        when a.Gov = 'Gov' then 0
        when a.Gov = 'Non-Gov' then 1
 else 99
end as Gov
```

FIGURE 4.1: *Code showing text to integer conversion.*

The second step is to group the fraud status of the DOs. There are three groups of DOs present in the data, they are broken into: DOs that are not fraudulent, DOs that are fraudulent and DOs that have not been classified into either group. In Table 4.2, the proportion of each group of the total sample are shown.

Grouping (Y)	DO Source	DO Source Description	Observations (n)	Proportion of Sample
2	[Unknown]	DOs originating from **unknown** sources/businesses	n=1,366,463	27%
1	[Fraudulent]	DOs originating from **fraudulent** sources/businesses	n=1,413,920	28%
0	[Non-Fraudulent]	DOs originating from **legitimate** sources/businesses	n=2,219,617	45%

TABLE 4.2: *Table showning the overall DO grouping based on fraud status.*

Referring to Table 4.2, the feature selection and model application will be done using the fraudulent and non-fraudulent groups.

DOs with a binary response of Y=1 are fraudulent. The classification methodology described in this chapter distinguishes the fraudulent DOs from the legitimate DOs. The model will be trained and tested using this as the outcome variable.

A common issue with analysing fraud data is the fact that the proportion of fraudulent observations is usually substantially lower than the proportion of non-fraudulent observations. Itoo *et al.* [30] propose the use of under-sampling the non-fraudulent cases in order to balance the data. In this study, different proportions of fraudulent and non-fraudulent observations are extracted, which are not necessarily representative of the actual proportions. If the proportion of fraudulent to non-fraudulent is considered on a weekly basis for the full data set, see Figure 4.2, it is

clear that the fraudulent observations constitute a smaller proportion of the total observations.

Referring to Figure 4.2, the proportions represented by fraudulent and non-fraudulent are approximately 50%-50% in week 1 of 2019 and moves closer to 25%-75% in the last few weeks of 2019. The decrease in fraudulent cases was as a result of efforts made by banks to curb the effects of fraud in the DO system throughout 2019 (see Section 2.7 for more detail).

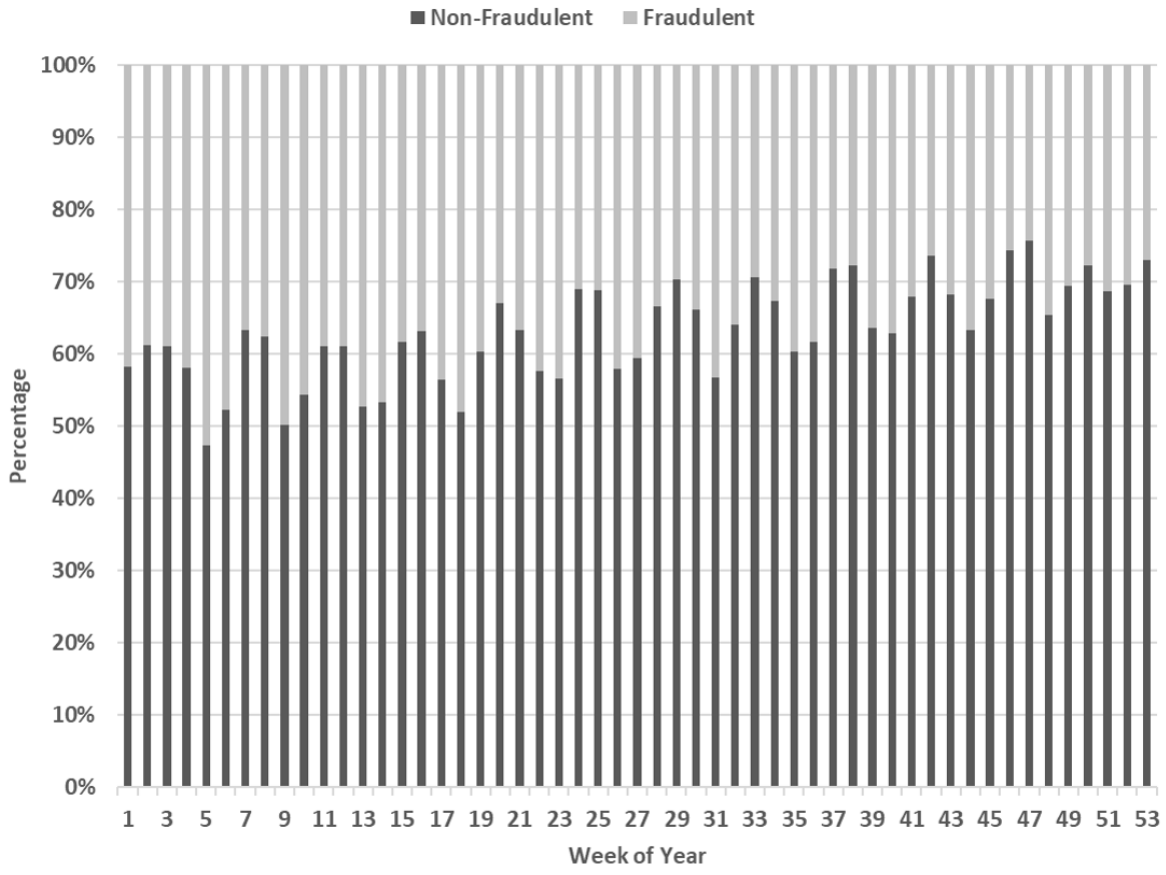

FIGURE 4.2: *Proportion of fraudulent and non-fraudulent observations per week during 2019.*

Due to the changing nature of the proportions of fraudulent and non-fraudulent observations, it is necessary to analyse the data by selecting samples with different proportions. The proportions of fraudulent to non-fraudulent observations to be considered are: 50%-50%, 80%-20% and 20%-80%. Refer to Figure 4.3 for a demonstration of how the proportions of fraudulent to non-fraudulent observations are selected.

The selection of proportional fraudulent and non-fraudulent observations is part of the data preparation step of the modelling procedure. This step is performed in the extraction phase of the procedure as data is imported into the R environment.

The data preparation step is critical as an input to the modelling process so that the model uses a cleaned and verified data set.

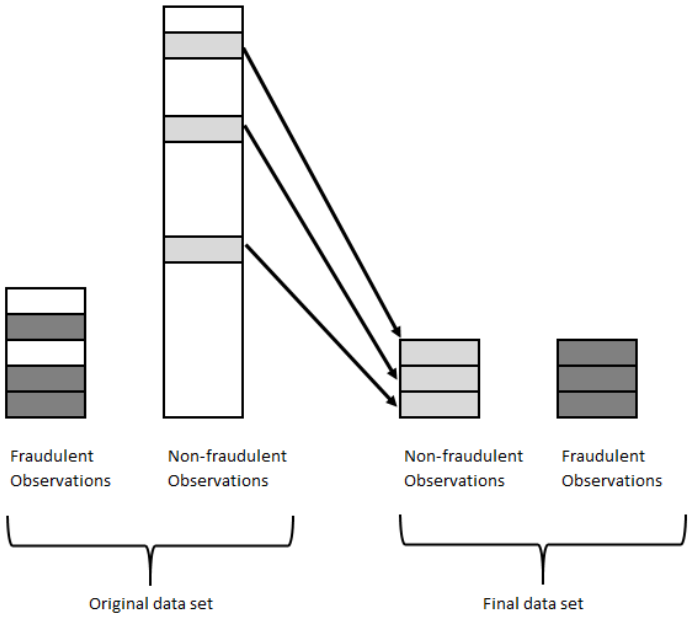

FIGURE 4.3: *Random under-sampling demonstration [30].*

The summarised preparation and extraction steps are (see Section A.1.2 in the appendix):

1. prepare data for extraction from the SQL server;
    - create numerical groupings for string character groupings;
    - create binary variables as described in Table 4.2;
2. connect the SQL server to R;
3. import proportions of fraudulent to non-fraudulent randomly selected observations into R.

Once the data is extracted the transformation procedure is conducted. The following conditions are part of the transformation procedure and are performed in the R environment.

4.2.2 Multicollinearity

Both LR and SVM require certain conditions to be met before these models can be applied. Multicollinearity is the first condition to be discussed, which affects both LR and SVM.

The VIF threshold of 5, together with equation (3.8), were used since any value larger than 5 is considered to be collinear and was removed [31].

The summarised steps of the process are shown below (see Section A.1.3 in the appendix):

1. import the necessary function to perform multicollinear analysis;
2. specify the VIF level to 5;

3. run the function, the basic steps of the procedure are;
    - produce a result on all features;
    - remove the variable with the largest VIF;
    - rerun the model using the decreased data set;
    - this process is repeated until a result is produced where none of the variables have a VIF above the threshold of 5.

4.2.3 Conditions for LR

The condition of sample size adequacy needs to be met, using equation (3.9). The number of actual observations in the data set ideally needs to be larger than this minimum. The outcome of the equation is discussed in more detail in the following chapter.

The condition of linearity of continuous input (X) variables needs to be inspected. Inspection is done by transforming the output (Y) variable using the logit function, equation (3.7), and then comparing the relationship of each continuous input (X) variable with the logit of the output (Y) variable. The relationship should be approximately linear.

4.2.4 Conditions for SVM

The final condition for SVM is that of normalised data which is achieved through a transformation once multicollinearity removed. This is to remove the influence of scale in variables on the model and is achieved by applying equation (3.12) to the data.

4.2.5 Training and Test Set Split

After the above conditions are met the data is then split into two subsets: the test set and the training set.

The size of the training and test set is specified as a percentage. Itoo *et al.* [30] make use of a training to testing data set split of 70%-30%. In this research an 80%-20% training to testing data split is used. For example, if the sample has $n = 500,000$ observations, the training set will consist of 400,000 observations (80% of the sample) and the test set will consist of 100,000 observations (20% of the sample).

Figure 4.4 shows the final division of the data set. The population (Database) is arbitrarily set at 5,000,000 fraudulent and non-fraudulent observations. From the 5,000,000 observations in the population (Database) fraudulent and non-fraudulent observations are randomly selected at the proportions shown in Figure 4.4. The final table is combined and from this table the training and test sets are selected.

Referring to Figure 4.4, the sample proportions are labelled as:

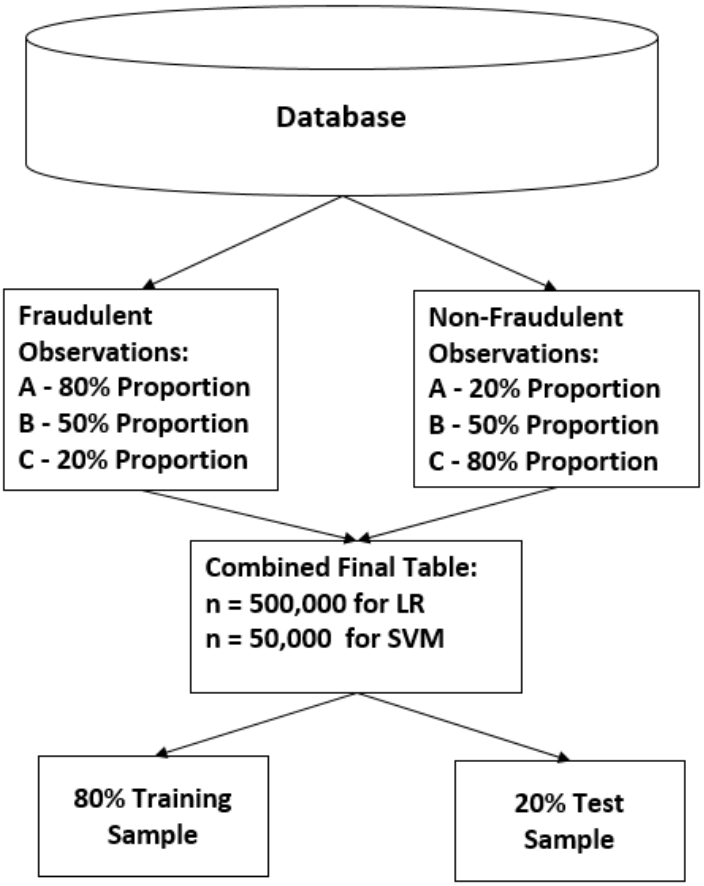

FIGURE 4.4: *Division of data set.*

- A - 80% fraudulent and 20% non-fraudulent observations;
- B - 50% fraudulent and 50% non-fraudulent observations; and
- C - 20% fraudulent and 80% non-fraudulent observations.

For the sake of simplicity, the sample proportions will be referred to as sample proportions A,B and C.

The test set will ultimately be used as the source of data for the data validation component of the analysis. However, before this is done feature selection is applied to both test and training sets to reduce the number of features in the data.

4.2.6 Feature Selection

More features included in the model does not guarantee a better model fit. Some variables may not be related to the output variable and will lead to deterioration of the fit of the model [31]. For this reason feature selection is performed, to remove the variables that are not related to the output variable. A positive outcome of the feature selection process is the decrease in input variables which requires less computation power for modelling.

The first feature selection method to be applied is the stepwise algorithm. This is done once all transformations are completed and conditions are met for both classification methods. The process discussed in Chapter 3 is used to perform stepwise feature selection.

Before PCA feature selection is done the training data is standardised according to equation (3.12) and sampling adequacy is tested using the KMO statistic, see equation (3.10). Once the data is standardised and the sample is adequate (KMO > 0.7) the PCA feature selection is applied.

A summary of the feature selection process is described below (see Sections A.1.6 and A.1.8 in the appendix):

1. the necessary function is loaded into R;
2. the training data is fed into the function;
3. both stepwise and PCA feature selection methods are applied according to the steps discussed in Chapter 3;
4. only the features selected as per the training data set are used when applying the classification models.

At this point the feature selection process is complete and the classification models can be applied.

4.3 Modelling Application

The model application phase of the machine learning process is described below. An overview of the process can be found in Figure 3.1.

At this point of the modelling process the data is prepared and the necessary conditions are met and the LR and SVM models can be applied.

The above mentioned classification models are chosen since they are strong linear classifiers and are commonly used in fraud detection situations, refer to Section 3.4. The RBF kernel will be used in the SVM models and the kernel can be either a linear or non-linear classifier which allows for model versatility [28].

The basic steps for the model application process are (see Sections A.1.4, A.1.7 and A.1.9 in the appendix):

1. import the necessary functions;
2. set the parameters in the function, this includes assigning the necessary training data set;
3. run the classification function;
4. a trained classification model is produced;

5. the test data is fed into the trained classification model;

6. several results are produced from fitting the model to the test set.

The above steps are applied to the training data set and the classification model is trained/created. The training data is 80% of the total data set, leaving 20% for the testing component of the data set. The test data is fed into the model after which the model predicts the binary output of the test set. The performance of the model is measured by comparing actual values to predicted values of the test set.

The results produced by the classification model on the test set are analysed in the next component.

4.4 Modelling Validation

The model validation component is centered around the classification accuracy of each permutation of feature selection method and classification method. Several permutations are assessed combining all feature selection methods with all classification methods.

The six permutations, per sample proportion, are:

1. no feature selection (all features with acceptable VIF scores are included) with logistic regression;

2. stepwise feature selection with logistic regression;

3. PCA feature selection with logistic regression;

4. no feature selection (all features with acceptable VIF scores are included) with SVM with an RBF kernel;

5. stepwise feature selection with SVM with an RBF kernel;

6. PCA feature selection with SVM with an RBF kernel.

With each permutation, a result is created which will be compared using the same set of measures (see Section A.1.5 in the appendix). The specific classification metrics are accuracy, sensitivity, specificity and the F1-measure. The equations for these metrics can be found in Table 3.2.

The ROC curve is used to analyse the goodness of fit for all permutations mentioned above. A good classification model will hug the left and top boundaries with as large an AUC as possible. This means that a large percentage of observations were classified correctly and consequently that the classification error rates are low, refer to Figure 3.4.

In conjunction with the ROC curve, the goodness-of-fit for the logistic regression permutations are assessed by the Hosmer-Lemeshow test (using hypothesis test (3.3)) and the McFadden's

Pseudo R^2 is applied for predictive capability. The tests are performed on all permutations of the logistic regression models. Refer to Chapter 3 for additional details on these diagnostic tests.

In addition to the above-mentioned diagnostic tests, AIC (using equation (3.20)) is used to assess the relative performance of the model permutations. The lower the AIC the better the performance of the model. The AIC is used as a relative performance measure for each model permutation.

Another method, used in conjunction with the above-mentioned measures, is to measure the calculation time of each of the models. Some models are mathematically more complex and this results in higher computation times. The larger a data set grows the higher the computation time becomes as well. For this reason the computation time will also be included in the results and used for comparison of the models.

This chapter describes the process the data goes through from the source system to where it is modelled. The process begins with data preparation, followed by data condition assessment. Feature selection is then applied to the data after which the models are applied. The final step in the ML process is to validate the model, which is done using classification accuracy metrics and post application diagnostics.

The next chapter will show the results produced by applying the above process to the DO data set.

CHAPTER 5

Results

Contents

The aim of this research was to investigate the success of LR and SVM in identifying fraudulent debit orders in the NPS. The results in this chapter were produced using the methodology specified in the previous chapter.

The results obtained from applying LR are presented in Section 5.1. In Section 5.2, the results from applying SVM are presented. Finally, in Section 5.3, the results obtained by the two methods are compared.

5.1 Logistic Regression

The first set of results discussed are the conditions that need to be met prior to the application of LR, followed by the performance and diagnostics checks.

5.1.1 Descriptive Statistics

The sample size used for testing is n=500,000 consisting of 63 independent (X) variables and 1 dependent (Y) variable (see Table 4.1 for more details). All observations are randomly selected

from the population of 5,000,000 observations generated during 2019. The training and test set sizes (as per the 80%-20% split) are 400,000 and 100,000 respectively. The list of independent variables shrinks once collinear variables are removed as shown later in the research and the final list of variables considered is shown in Table 5.2.

There are several non-ordinal categorical variables that need to be transformed into dummy variables. *Province* is an example of this. The total fraudulent cases per province are ranked from most to least fraudulent cases, refer to Table 5.1. The province with the highest number of fraudulent cases is identified as the reference province and the remaining provinces are encoded as dummy variables. In this case it is "Head Office", this is due to the fact that many fraudulent cases are proactively identified at the head office. For example, when the fraudulent DO is identified in Limpopo the dummy variable for Limpopo will be classified as 1, and the dummy variable for Limpopo will be classified as 0 for all observations in the other nine provinces.

There are several independent variables for which this process was followed including: *Official Client Type*, *Quality Banking Client Description*, *Title* and *Stable Product Usage*. Due to the sensitive nature of these variables, volumes of fraudulent DOs will not be shown. The remaining variables from Table 5.2 are either continuous, binary or categorical (ordinal) values and did not need to be transformed.

Province Code	Fraudulent Cases
Head Office	882,625
Province Gauteng	149,134
Province Kwa-Zulu Natal	76,739
Province Western Cape	71,354
Province Eastern Cape	64,976
Province Mpumalanga	49,772
Province Limpopo	35,362
Province North West	35,289
Province Free State	27,037
Province Northern Cape	21,632

TABLE 5.1: *Total fraudulent cases per province.*

5.1.2 Conditions

All conditions specified in the methodology (Chapter 4) were assessed, including multicollinearity, variable linearity, sample size adequacy and variable standardisation.

Multicollinearity

Using equation (3.8) on the full sample (n=500,000), the first condition that is assessed is the removal of multicollinearity. Multiple derived variables were present in the DO data which caused multicollinearity. These variables were removed. The initial list of variables, using a

VIF threshold of 5, is decreased to 31, excluding the dependent variable. The VIF values of the variables can be seen in Table 5.2. The largest VIF value is for *Ave Inflows 6 Month Grouped* with a value of 3.83.

The independent variables that remain after multicollinearity checks are used in all of the remaining processes. This means that the variables identified in Table 5.2 are used by all LR and SVM models.

TABLE 5.2: *Remaining variables and VIF scores.*

Begin of Table 5.2	
Variable Name	VIF Score
Age band	1.46
App Registered	1.53
Banking Client	1.82
Banking Client New	1.03
Employee	1.05
Credit Card Client	1.44
Fixed Savings Client	1.02
Flexible Savings Client	1.11
Gov	1.23
Grouping DebiCheck Branch Dispute	1.00
Inflow Current Month Grouped	3.43
Official Client Type Fee	3.17
Official Client Type New	1.01
Province Eastern Cape	1.20
Province Free State	1.08
Province Gauteng	1.37
Province Kwa-Zulu Natal	1.20
Province Limpopo	1.10
Province Mpumalanga	1.12
Province North West	1.12
Province Northern Cape	1.06
Province Western Cape	1.21
R45Flag	1.46
R99Flag	1.03
Reverter	1.33
Reverter New	1.12
Risk Group NLR Compuscore	1.17
Term Loan Client Good Standing	2.92
Title DR	1.00
Title MADAME	1.02
Title MISS	1.21
Title MRS	1.20
Title MS	1.17
Title PROF	1.01
Val group	1.80
Ave Inflows 6 Month Grouped	3.83
Branch Visits 12 Month Grouped	1.33
POS Current Month Grouped	1.54
POS Value Grouped	1.06
Ave DO 6 Month	2.54
DO Current Month Grouped	2.20
Num ATM Withdrawals Current Month Grouped	2.11

Continuation of Table 5.2	
Variable Name	VIF Score
ATM Withdrawal Amount Grouped	2.26
Quality Banking Client	2.25
Stable Product Usage Stable	1.26
Stable Product Usage Unstable	1.33
End of Table 5.2	

Variable linearity

The next condition to be considered is variable linearity. Referring to Table 5.2, the independent variables that remain after multicollinear variables are removed show that there are no continuous variables remaining. All variables are either binary or ordinal categorical; therefore, it is not necessary to visually inspect the logit relationship. Since the visual inspection of linearity is not necessary for the remaining variables it is possible to move onto the next condition.

Sample size adequacy

The final condition to consider for logistic regression is sample size adequacy. In the case of LR, this condition is assessed using equation (3.9):

$$n = 100 + 50k = 100 + 50(31) = 1,650. \tag{5.1}$$

The sample size used for testing is 500,000 which is split into a training set (n=400,000) and a test set (n=100,000); therefore, the sample size is adequate to apply LR.

The above conditions are met and the data set can be used for the feature selection component of the modelling process.

Standardisation and sampling adequacy conditions for PCA

Having assessed all necessary conditions for application of LR to the data, the conditions for application of PCA needs to be considered.

The first condition that needs to be assessed is sampling adequacy for application of PCA. Sampling adequacy is assessed using the KMO test. A value of 0.7 or above, is desirable for the KMO test [21]. Performing the test on the sample size of 500,000 the overall metric achieved is 0.81. This overall KMO value is well above the minimum value of 0.6 and above the 0.7 minimum required for this research. Table 5.3 shows the KMO statistic for all variables. Only variables with individual KMO scores greater than 0.7 were considered. There was a significant drop in the number of variables when compared to the number of variables listed in Table 5.2.

Variable	KMO Value
Overall KMO Score	0.81
Age band	0.73
Banking Client	0.77
Credit Card Client	0.88
Flexible Savings Client	0.87
Gov	0.87
Inflow Current Month Grouped	0.84
Reverter	0.71
Ave Inflows 6 Month Grouped	0.84
Branch Visits 12 Month Grouped	0.85
POS Current Month Grouped	0.78
Ave DO 6 Month	0.78
DO Current Month Grouped	0.77
Quality Banking Client	0.81

TABLE 5.3: *KMO statistic for the remaining independent variables.*

The final condition that needs to be met for application of PCA is the standardisation of variables, since PCA is affected by the scale of the input variables. The standardisation of input variables is done by applying equation (3.12) to each variable. This is done in the preparation phase, before PCA is applied.

PCA and stepwise feature selection is then applied to the cleaned data.

5.1.3 Model Performance

All conditions were met as presented above. LR could therefore be applied to the selected data set. The R code used to apply both models can be found in Appendix A. The necessary metrics to assess the performance of each LR model are discussed.

The general performance comparison considers the information contained in Table 5.4. The three models, per sample proportion A to C, as shown in Table 5.4, are:

- LR with no feature selection (all features with acceptable VIF scores are included);
- LR with stepwise feature selection; and
- LR with PCA feature selection.

AIC

The AIC gives an indication of the simplicity of a model. The smaller this value the better the fit of the model and the simpler the model is.

[1]These values are less than the lower limit for the McFadden pseudo R^2

Sample Proportion	Feature selection method	Accuracy	AIC	Hosmer-Lemeshow p-value	McFadden pseudo R^2
A	No feature selection	83.48%	91,540	2.20E-16	0.2385
A	Stepwise feature selection	83.45%	91,390	2.20E-16	0.2394
A	PCA feature selection	79.89%	116,419	2.20E-16	0.0301 [1]
B	No feature selection	72.77%	420,700	2.20E-16	0.2414
B	Stepwise feature selection	72.85%	86,730	2.20E-16	0.2480
B	PCA feature selection	59.48%	161,227	2.20E-16	0.0310 [1]
C	No feature selection	82.78%	93,640	2.20E-16	0.2205
C	Stepwise feature selection	82.84%	94,120	2.20E-16	0.2161
C	PCA feature selection	79.99%	116,989	2.20E-16	0.0261 [1]

TABLE 5.4: *Logistic regression summary table.*

The AIC values vary substantially for the LR models. The largest AIC value is for the sample proportion B with no feature selection model with AIC=420,700. The smallest AIC is for the sample proportion B with stepwise feature selection model with AIC=86,730.

Hosmer-Lemeshow test

Hosmer-Lemeshow tests the goodness-of-fit of the model. The test shows how well the model fits the data. Looking at the Hosmer-Lemeshow statistical significance, it is clear that there are issues with the data when used for LR modelling.

Referring to Table 5.4 the p-value produced by each model is below 0.05, and using the Hosmer-Lemeshow hypothesis test (3.3) from Chapter 3, the H_0 hypothesis should be rejected; therefore, sufficient evidence exists to suggest that the model does not fit the data. This result is expected as the sample size exceeds the suggested maximum sample size of n = 25,000 and this test is actually not a reflection on the fit.

The above-mentioned Hosmer-Lemeshow result does not provide any additional information about the goodness-of-fit. The following information gained from the Hosmer-Lemeshow test provides additional insight.

It is interesting to note the behaviour of the Hosmer-Lemeshow test regardless of all the models having sample sizes greater than n=25,000. Referring to Figures 5.1 to 5.3, these graphs show

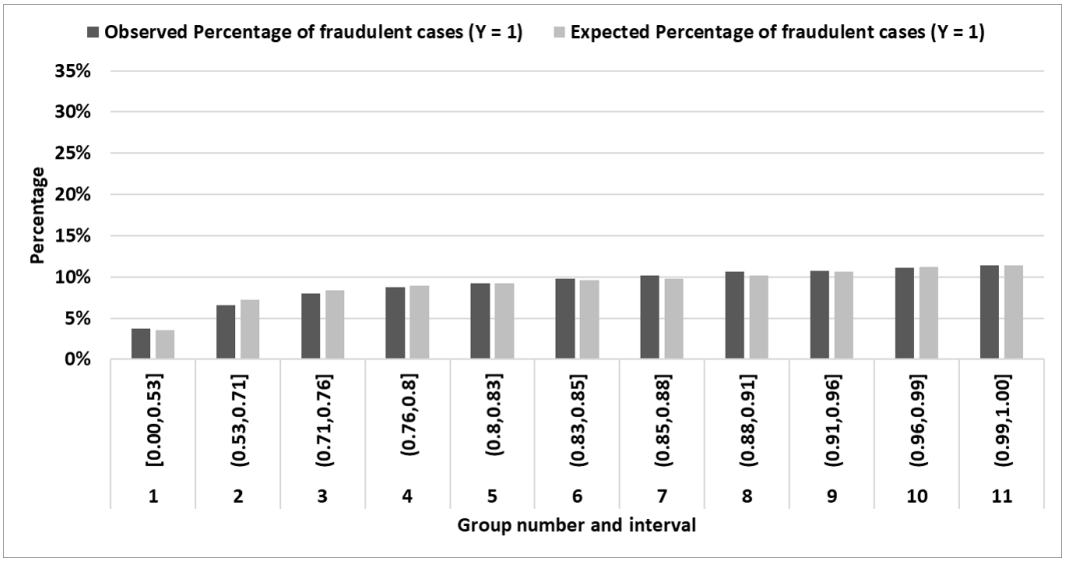

FIGURE 5.1: *The observed and expected percentage cases per group for the fraudulent cases, generated when applying the Hosmer-Lemeshow test (sample proportion A with stepwise feature selection).*

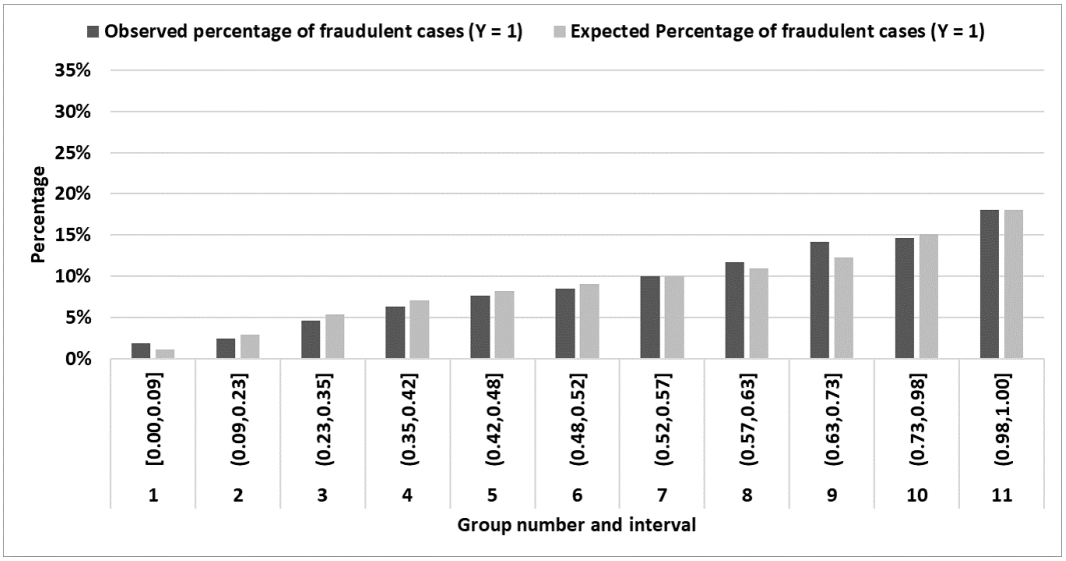

FIGURE 5.2: *The observed and expected percentage cases per group for the fraudulent cases, generated when applying the Hosmer-Lemeshow test (sample proportion B with stepwise feature selection).*

the percentage of total observations classified into 11 approximately equal sized groups. These groups are in ascending order of the predicted probability that a case is fraudulent, given that the actual outcome is fraudulent, or the probability that $\hat{Y} = 1$, given that $Y = 1$. The actual percentage of observations is compared to the expected percentage of observations showing how well the model fits the data.

Figure 5.1 shows the results for sample proportion A, sampled to contain 80% fraudulent cases. In this case, the cut-off probability for identifying fraudulent cases is encompassed in the first interval. Therefore, only between 0% and 5% non-fraudulent cases and therefore more than 95% fraudulent cases with $P(\hat{Y} = 1) > 0.5$, are identified. The sensitivity of this application, defined as the probability that the predicted outcome is fraudulent, given that the actual outcome is

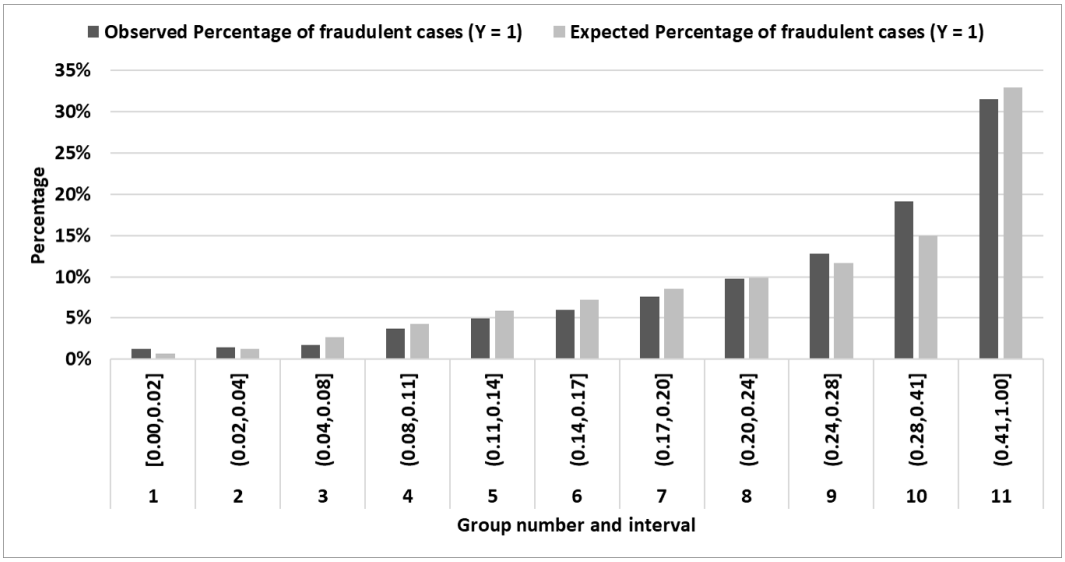

FIGURE 5.3: *The observed and expected percentage cases per group for the fraudulent cases, generated when applying the Hosmer-Lemeshow test (sample proportion C with stepwise feature selection).*

fraudulent, is 97.64% (see Table 5.5). The expected and observed frequencies are approximately the same.

Figure 5.2 shows the results for sample proportion B, sampled to contain 50% fraudulent cases. The boundaries of the 11 intervals are of approximately equal size, and the cut-off probability for identifying fraudulent cases is encompassed in the sixth interval. Approximately 70% of the observations were identified as fraudulent. The sensitivity, as in Table 5.5, is 72.86%. Observed and expected frequencies are approximately the same.

Figure 5.3 shows the results for sample proportion C, sampled to contain 20% fraudulent cases. The largest percentage of observations fall into the eleventh group. The eleventh group has boundaries that encompass the cut-off probability of 0.5 used for this research, if $P(\hat{Y} = 1) > 0.5$, then the case is flagged as fraudulent. Between 30% and 35% of the observations in the eleventh group have a probability of at least 0.41 of being fraudulent, and are actual fraudulent cases. The sensitivity of the test is calculated using the cut-off probability of 0.5, and the calculated value (see Table 5.5) is 27.85%. The expected and observed percentages are approximately the same.

Figures 5.1 to 5.3 show the large effect that sample proportion can have on sensitivity and the model fit in general. Sample proportion A and C appear to be extreme. Sample proportion B (50%-50%) is producing moderate results that are not so extreme.

McFadden's Pseudo Coefficient of Determination

McFadden's Pseudo R^2 test assesses the predictive capability of the model. The test gives an indication of how well the independent variables can predict the dependent variable. Using the McFadden's Pseudo R^2 test from Section 3.2.4, it is clear that all models, except the PCA feature selection models, have high predictive power since the diagnostic values fall between

0.2 and 0.4 [39]. This means that PCA LR models do not have high predictive capability (see Table 5.4).

The diagnostic measures above show that no feature selection and stepwise feature selection models have high predictive capability and that all of the models have sample sizes that are too large to use Hosmer-Lemeshow statistic to determine goodness-of-fit.

Table 5.5 shows the classification performance metrics of the LR models.

Sample Proportion	Feature selection method	Accuracy	Sensitivity	Specificity	F1-Measure	Run Time (seconds)
A	No feature selection	83.48%	97.76%	26.39%	62.08%	6.81
A	Stepwise feature selection	83.45%	97.64%	26.69%	62.17%	2.17
A	PCA feature selection	79.89%	99.99%	00.01%	50.00%	0.72
B	No feature selection	72.77%	72.58%	72.93%	72.77%	23.46
B	Stepwise feature selection	72.85%	72.86%	72.84%	72.85%	0.45
B	PCA feature selection	59.48%	65.11%	53.78%	59.44%	0.82
C	No feature selection	82.78%	27.71%	96.65%	62.18%	6.47
C	Stepwise feature selection	82.84%	27.85%	96.68%	62.26%	2.17
C	PCA feature selection	79.99%	00.01%	99.99%	50.00%	0.74

TABLE 5.5: *Summary of the performance measures of the LR models.*

Consulting Table 3.2 for equations, the metrics considered for specific model performance are: accuracy, sensitivity, specificity, F1-measure and run time. The value for each metric is summarised in Table 5.5.

Accuracy

Accuracy gives an indication of how many observations were predicted correctly compared to all predictions. This is a good metric to assess general model performance.

The highest performing models used sample proportion A and the lowest performing models used sample proportion B. The best model is the no feature selection model using sample proportion A with 83.48% accuracy. The PCA feature selection method for each model yields the lowest accuracy per sample proportion and will not be considered as a viable solution due to the other high performing models. The KMO values caused a large drop in the number of input (X)

variables resulting in a loss of information for the PCA models. See Table 5.3 for the KMO values for variables included in the PCA feature selection method.

Sensitivity and Specificity

Sensitivity represents the number of correctly predicted true positive values out of all positive values. With regards to fraudulent DOs that is: the number of correctly identified fraudulent DO cases out of all fraudulent cases.

From the point of view of fraud detection, sensitivity is a critically important metric. The ability to correctly identify fraudulent DOs with high certainty gives a lot of confidence in the ability of the model to serve its purpose of identifying fraud. The sensitivity of the models with sample proportion A are the highest, reaching close to 100%. The lowest sensitivities are produced where sample proportion C is used.

The specificity of the model shows how accurately the model is able to identify negative cases, in the case of DOs that is how accurately the model is able to identify non-fraudulent cases. This is not a cause for major concern since the purpose of the model is to focus around predicting fraud and not predicting non-fraud. There are models that have specificity close to 100% and other models where the specificity is closer to 20%.

It is interesting to note the sensitivity and specificity rates alternate between sample proportions A and C. That is when a model produces high sensitivity rates and low specificity rates for sample proportion A, it gives low sensitivity rates and high specificity rates for sample proportion C. This indicates that, for both sensitivity and specificity to be relatively high, the chosen sample proportion lies somewhere between sample proportion A and sample proportion C. The modeller should decide on the trade-offs between high sensitivity (sample proportion A), or a lower sensitivity, but an acceptable specificity (sample proportion B).

Run time

The run time metric considers the calculation time for training the LR model. This metric scales with the number of observations and independent variables used in the model and is a very important consideration in the real-world application of classification.

The longest run times was achieved when no feature selection was applied, with significant drop-offs in run time for the stepwise and PCA feature selection models. This drop-off is due to fewer independent variables being included in the model. Although there are large differences in run time for LR models, this metric will be used primarily to compare LR and SVM.

ROC

Table 5.5 summarises some of the most important model performance metrics. The graph in Figure 5.4, shows the ROC curve for the highest performing model using stepwise feature

selection and LR as a classifier. Sample proportion A was used to produce this result.

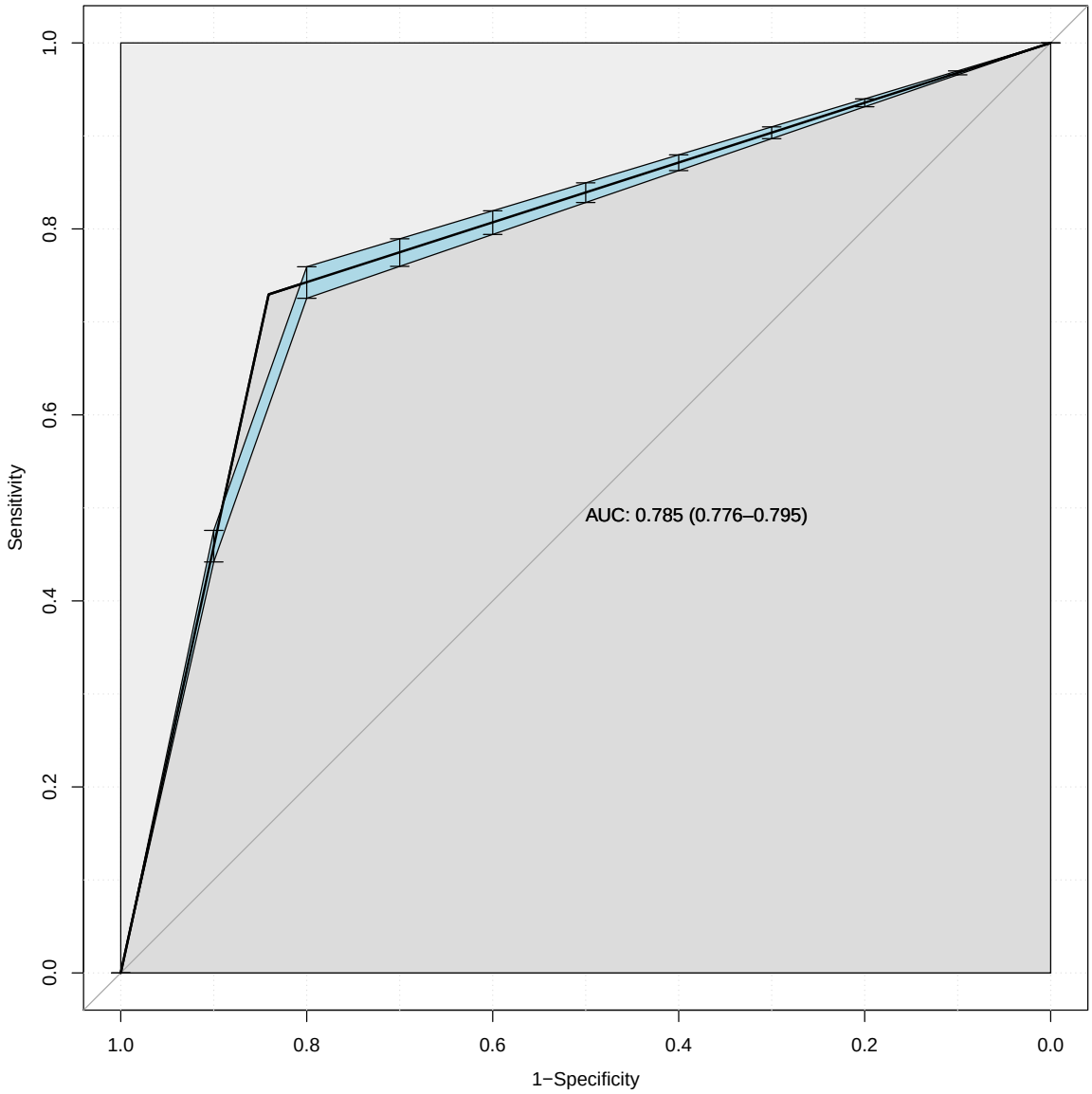

FIGURE 5.4: *Logistic regression stepwise ROC (sample proportion A with stepwise feature selection).*

The graph in Figure 5.4, shows the different sensitivity and specificity combinations at different thresholds of posterior probability. The curve should ideally hug the top and left boundaries as close as possible. This increases the area under the curve, which is indicative of higher prediction accuracy. The ROC curve is well above the 45° line with an AUC of 78.50%. The light blue area is the confidence interval for the ROC curve. This can be considered a high performing result and will be used in the comparison to the SVM models.

Independent variable interpretation

The odds ratios for all sample proportions with stepwise feature selection LR models are shown in Tables 5.6 to 5.8.

Variable	Odds Ratio	Significance
Age band	1.0631	<0.0001
Banking Client	0.9008	0.0054
Banking Client New	0.9342	0.3889
Employee	0.5909	0.0002
Fixed Savings Client	0.9207	0.0136
Inflow Current Month Grouped	1.0404	0.0007
Official Client Type New	0.309	0.0298
Province Eastern Cape	0.8349	<0.0001
Province Limpopo	0.7757	<0.0001
Province Northern Cape	1.1267	0.0663
R45Flag	69.2835	<0.0001
R99Flag	36.2350	<0.0001
Risk Group NLR Compuscore	1.1389	<0.0001
Title MS	0.9241	0.0007
Title PROF	1.1246	0.9945
Val group	0.7294	<0.0001
Ave Inflows 6 Month Grouped	1.1485	<0.0001
Branch Visits 12 Month Grouped	0.9373	<0.0001
POS Current Month Grouped	1.0591	<0.0001
Ave DO 6 Month	0.9347	<0.0001
Num ATM Withdrawals Current Month Grouped	0.966	<0.0001
Quality Banking Client	0.8342	<0.0001

TABLE 5.6: *Variable odds ratio and significance (sample proportion A with stepwise feature selection).*

Referring to Table 5.6, the odds ratios for *Banking Client New*, *Province Northern Cape* and *Title PROF* are not significant. The remaining variables have significant odds ratios.

Referring to Table 5.6, the odds ratio of *R99Flag* is 36.2350. An odds ratio of 36.2350 for *R99Flag* means that the odds of a fraudulent transaction is 36.2350 more if *R99Flag* = 1 than when *R99Flag* = 0 (the reference). The value is significant because the p-value for the *R99Flag* coefficient is less than 0,05. This ties in with the "R99 scam" identified in Section 2.7.

If the *Age band* increases by 1 unit, the odds of the DO being fraudulent increases by 1.05, showing a positive relationship between age and the likelihood of being exposed to fraud. These *R99Flag* and *Age band* variables tie in with the example of the "R99 scam" given in Section 2.7, and it also shows that elderly persons are more exposed to these scams.

Referring to Table 5.6, the odds ratio for the *Employee* variable is 0.5909. The odds of a DO being classified as fraudulent (DO Classification = 1) is decreased by a factor of 0.5909. If a person is an employee of the bank.

Referring to Table 5.6, odds ratios of above 1 are considered risk factors. *Age band* has an odds ratio of 1.0631 and is significant, which means that the older an individual is the more likely it

Variable	Odds Ratio	Significance
Age band	1.055	<0.0001
Banking Client	0.9136	0.0195
Banking Client New	1.2907	0.0011
Employee	0.5005	<0.0001
Fixed Savings Client	0.9689	0.3370
Inflow Current Month Grouped	1.0594	<0.0001
Official Client Type New	0.329	0.0362
Province Eastern Cape	0.8746	0.0002
Province Limpopo	0.818	<0.0001
Province Northern Cape	1.1773	0.0101
R45Flag	53.0182	<0.0001
R99Flag	31.6643	<0.0001
Risk Group NLR Compuscore	1.117	<0.0001
Title MS	0.958	0.0631
Title PROF	25.6118	0.0065
Val group	0.7189	<0.0001
Ave Inflows 6 Month Grouped	1.1645	<0.0001
Branch Visits 12 Month Grouped	0.9367	<0.0001
POS Current Month Grouped	1.0434	<0.0001
Ave DO 6 Month	0.9388	<0.0001
Num ATM Withdrawals Current Month Grouped	0.9636	<0.0001
Quality Banking Client	0.8351	<0.0001

TABLE 5.7: *Variable odds ratio and significance (sample proportion B with stepwise feature selection).*

Variable	Odds Ratio	Significance
Age band	1.0361	<0.0001
Banking Client	0.9459	0.1619
Banking Client New	1.0042	0.9565
Employee	0.5713	<0.0001
Fixed Savings Client	1.0531	0.1134
Inflow Current Month Grouped	1.0586	<0.0001
Official Client Type New	0.564	0.2043
Province Eastern Cape	0.8791	<0.0001
Province Limpopo	0.9151	0.0574
Province Northern Cape	1.1204	0.0631
R45Flag	70.4372	<0.0001
R99Flag	29.6274	<0.0001
Risk Group NLR Compuscore	1.1308	<0.0001
Title MS	0.9125	<0.0001
Title PROF	0.1322	0.0164
Val group	0.7239	<0.0001
Ave Inflows 6 Month Grouped	1.1437	<0.0001
Branch Visits 12 Month Grouped	0.9357	<0.0001
POS Current Month Grouped	1.0527	<0.0001
Ave DO 6 Month	0.9428	<0.0001
Num ATM Withdrawals Current Month Grouped	0.9799	0.0015
Quality Banking Client	0.8388	<0.0001

TABLE 5.8: *Variable odds ratio and significance (sample proportion C with stepwise feature selection).*

is that they will fall victim to DO fraud. Odds ratios below 1 are considered to be protective factors. *Val group* indicates the value of a DO. The odds ratio for *Val group* is 0.7294 and is significant, which means the higher the value of a DO is, the less likely it is to be fraudulent. This

makes intuitive sense since fraud on large DOs will attract more attention to it and remaining unnoticed is a priority for DO fraud.

When comparing the odds ratios for the different sample proportions, refer to Tables 5.6 to 5.8, there are small differences. In general, the odds ratios remain either risk factors or protective factors and remain either significant or insignificant across the sample proportions. There are a few exceptions like *Banking Client New* which switches between significant and not significant. *Title PROF* switches between significance and changes from a risk factor to protective factor. This could be due to few observations where *Title PROF* has a value of 1.

The above section analysed the LR results, the following section will analyse the SVM results.

5.2 Support Vector Machine

The first component of this section includes conditions that need to be met in order to apply SVM, followed by the summary of the performance measures.

5.2.1 Descriptive Statistics

The SVM models were produced using a smaller sample of n=50,000 observations. This is due to the increased calculation time of each model. The variables shown in Table 5.2 are considered for the SVM modelling component. Using the 80%-20% training-to-test set split the subsets are n=40,000 for the training set and n=10,000 for the test set. All observations are randomly selected from the population of 5,000,000 observations which were recorded during 2019.

5.2.2 Conditions

There are two conditions that need to be met for SVM to be applied. One is the removal of collinear variables and the transformation (or standardisation) of the data. The same conditions for PCA need to be met before the feature selection method is applied.

Multicollinearity and standardisation

The same VIF method used in the LR results component is applied here. The same VIF threshold of 5 is used producing the same list of 31 input variables as seen in Table 5.2. These 31 variables are used for all the models to follow.

The input variables are standardised to remove the influence of scale. This is done by applying equation (3.12) once the multicollinearity is removed.

The above two conditions are met for SVM application. Feature selection is subsequently performed, before the SVM is applied.

Standardisation and sample size conditions for PCA

Having assessed all necessary conditions for application of SVM, the conditions for application of PCA feature selection are considered.

The two conditions that need to be met for application of PCA is sampling adequacy and the removal of scale from features. The same procedure is followed as with LR and further detail can be found in the LR section. The overall KMO statistic of 0.81 is achieved, this is sufficient to proceed with the model application. The remaining KMO values for individual variables can be found in Table 5.3. The second condition of data normalisation is achieved by applying equation (3.12), this is already done during the SVM condition process.

Since the above conditions are met the models are applied and the performance metrics are analysed.

5.2.3 Model Performance

The following section considers the classification performance metrics. A combination of all metrics are summarised in Table 5.9, which include confusion matrix deduced measures. Refer to Table 3.2 for the equations of accuracy, sensitivity, specificity and the F1-measure.

The three models, per sample proportion (sample proportion A to C), being considered in Table 5.9 are:

- SVM with no feature selection (all features with acceptable VIF scores are included);
- SVM with stepwise feature selection;
- SVM with PCA feature selection.

Accuracy, sensitivity and specificity

The highest accuracy produced by the SVM models is 85.22% where stepwise feature selection is done and sample proportion A was used. The lowest accuracy produced using sample proportion B with PCA feature selection at 57.99% accuracy.

When considering sensitivity the SVM classifier achieves a high score of nearly 100% in some models. The sensitivity of the sample proportion A models is very high whereas the sensitivity of the sample proportion C models is very low. The sample proportion B models sit close to 75% for the no feature selection and the stepwise feature selection models.

Some of the SVM classifier models achieve specificity scores of nearly 100%. The opposite of sensitivity rates is true for specificity. The specificity of the sample proportion A models is very low whereas the specificity of the sample proportion C models is very high. The sample

Sample Proportion	Feature selection method	Accuracy	Sensitivity	Specificity	F1-Measure	Run Time (seconds)
A	No feature selection	84.37%	99.65%	20.71%	60.18%	628
A	Stepwise feature selection	85.22%	98.12%	31.51%	64.81%	295
A	PCA feature selection	80.80%	99.99%	00.01%	50.53%	789
B	No feature selection	75.71%	75.40%	76.03%	75.71%	534
B	Stepwise feature selection	73.36%	72.21%	74.53%	73.37%	471
B	PCA feature selection	57.99%	52.79%	63.32%	58.15%	479
C	No feature selection	82.59%	25.32%	97.39%	61.36%	532
C	Stepwise feature selection	82.08%	22.59%	97.46%	60.02%	347
C	PCA feature selection	79.53%	00.03%	99.99%	50.19%	910

TABLE 5.9: *SVM summary table.*

proportion B models sit close to 75% for the no feature selection and the stepwise feature selection models.

ROC

The ROC for the highest performing SVM model is given in Figure 5.5. The AUC of this model is 88.2% with the ROC curve hugging the upper boundary more than the left boundary. This is indicative of the high level of sensitivity of the model. This ROC was produced using sample proportion A. Looking at Table 5.9 it is clear that these models produced very high sensitivity rates with low specificity rates.

Having looked at both LR and SVM classification results in isolation, it is necessary to consider these classification methods relative to one another.

5.3 Comparison Analysis

This section of the results compares LR and SVM classification methods directly. Figure 5.6 to 5.9, shows the accuracy, sensitivity, specificity and run time for the two classification methods relative to one another. Each feature selection method, that is no feature selection, stepwise feature selection and PCA feature selection, are also compared for both classification methods.

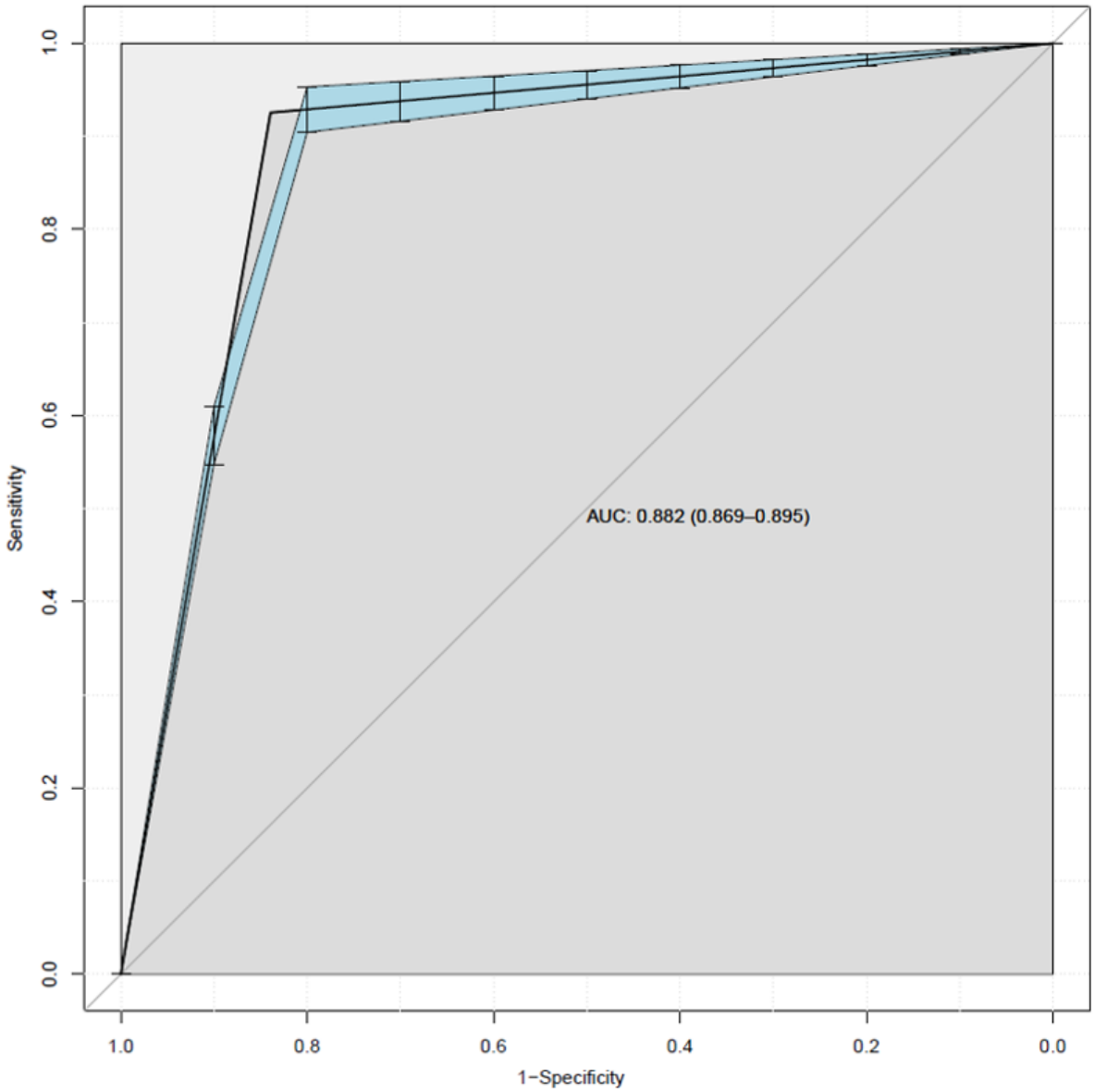

FIGURE 5.5: *SVM no feature selection ROC (sample proportion A).*

For each feature selection method and classification method there is also a sample proportion consideration which will be analysed.

5.3.1 Accuracy

Referring to Figure 5.6, there are some key observations to make regarding accuracy. For both LR and SVM the sample proportion that produces the highest accuracy is sample proportion A, followed by sample proportion C and then sample proportion B.

The models with sample proportion A produce approximately equivalent results for both classifiers. The highest accuracy that a single model produced was achieved using sample proportion A with an SVM model with no feature selection producing an accuracy of 85%.

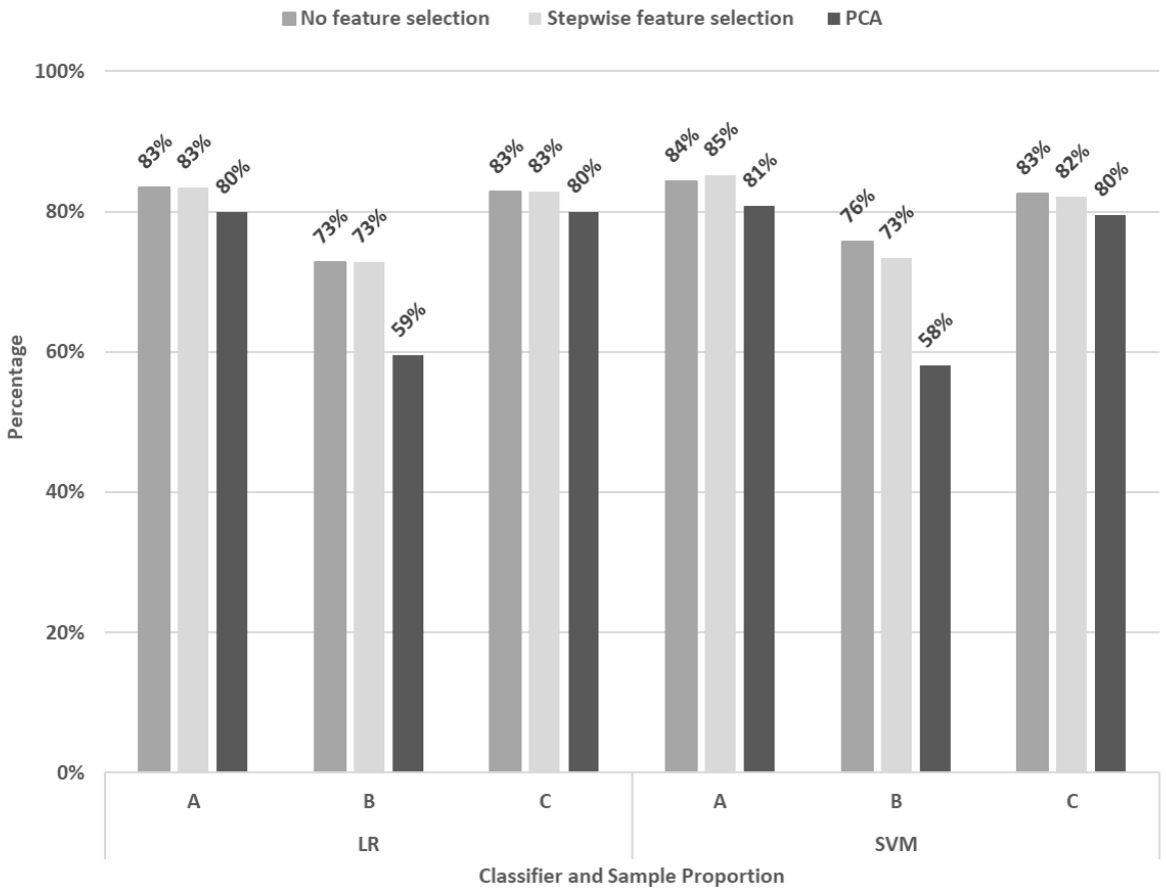

FIGURE 5.6: *Accuracy summary per classifier, sample proportion and feature selection method.*

Across all sample proportions, models and feature selection methods the lowest accuracy is produced by models where PCA feature selection is applied. In general, the accuracy rates produced by no feature selection and stepwise feature selection are approximately the same, indicating a limited positive impact by implementing feature selection.

5.3.2 Sensitivity and Specificity

Referring to Figure 5.7 for sensitivity comparison. Sample proportion A produces the highest sensitivity, followed by sample proportion B and then sample proportion C.

The sample proportion A SVM models produce a sensitivity just below 100%. This means that these models flag fraud correctly every time. This does come at a cost of very low specificity, producing large amounts of false positives (or false alarms). This is a problem experienced by other researchers, refer to Section 3.4.

Referring to Figure 5.8 for specificity comparison. The highest specificity rates were achieved using sample proportion C. These rates are just below 100%.

Take note of the specificity for the sample proportion A groups for both classification methods.

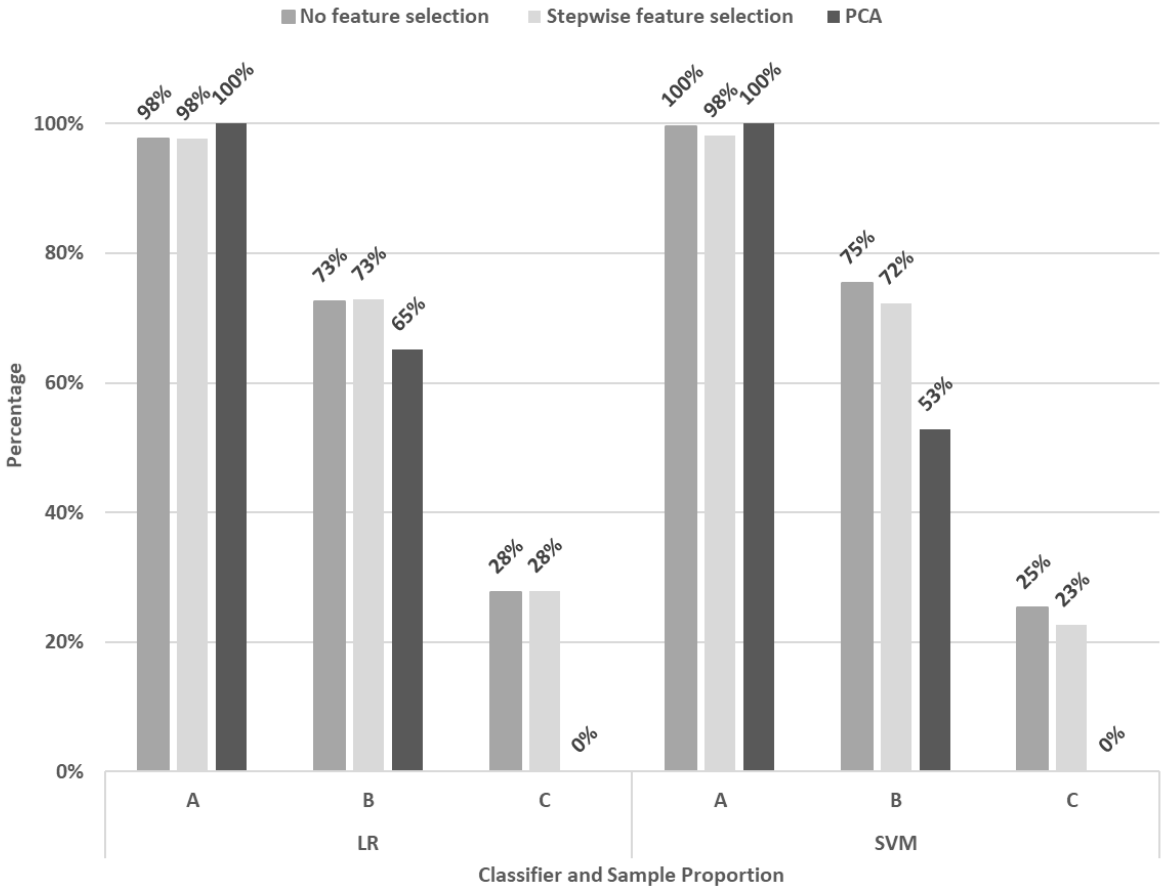

FIGURE 5.7: *Sensitivity summary per classifier, sample proportion and feature selection method.*

These specificity rates are very low at around 25%. This means that the sample proportion A has a high rate of true positive identification and a high rate of false positive identification. This produces a large degree of false positives as the model is over classifying.

For the sample proportion C models the reverse of the above is true. These models have high specificity and low sensitivity. This means the models are very capable of identifying non-fraudulent cases and will not readily classify a non-fraudulent case as fraudulent, but the models also struggle to identify fraudulent cases. These models are under classifying.

For accuracy, sensitivity and specificity there is minimal difference between LR and SVM. This is the same conclusion reached by Oza [49], who used the same classification methods to identify credit card fraud. There is a significant impact produced by the sample proportion used, there was a lot of focus placed on this problem by other researchers, refer to Section 3.4. Feature selection had a very small influence on the confusion matrix performance metrics; however, this may not be the case in the run time results.

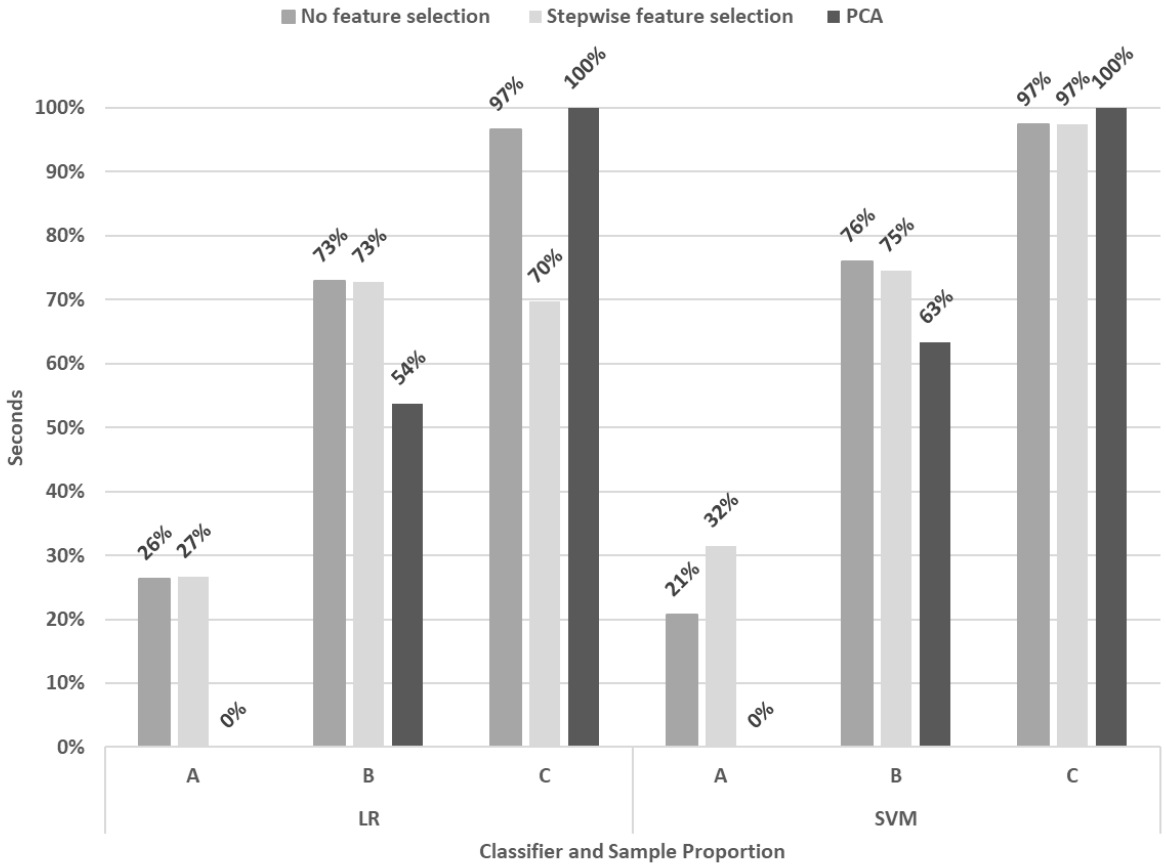

FIGURE 5.8: *Specificity summary per classifier, sample proportion and feature selection method.*

5.3.3 Run Time

Figure 5.9 shows the run time comparison. On average the SVM models run much longer than the LR models. This is due to the complexity of the SVM models and the more complicated mathematics result in longer calculation times.

The longest running models are the SVM models that use no feature selection. This is due to the increasing number of variables causing additional calculation time. The feature selection makes minimal impact on the accuracy, sensitivity and specificity of each model, but results in better run times.

From a practical point of view there is limited scope for application of SVM where data sets are very large due to the large calculation times. In fact, one of the strengths of SVM models is the high accuracy rates on small data sets, refer to Chapter 3 for further discussion on this. When the data sets become larger LR becomes more applicable due to the shorter calculation times.

The results section first considered that all conditions were met for all feature selection and classification methods. The conditions were met to a satisfactory level which means the feature selection and classification methods could be applied. The feature selection methods decreased the number of input variables from the initial list. After this the classification methods were

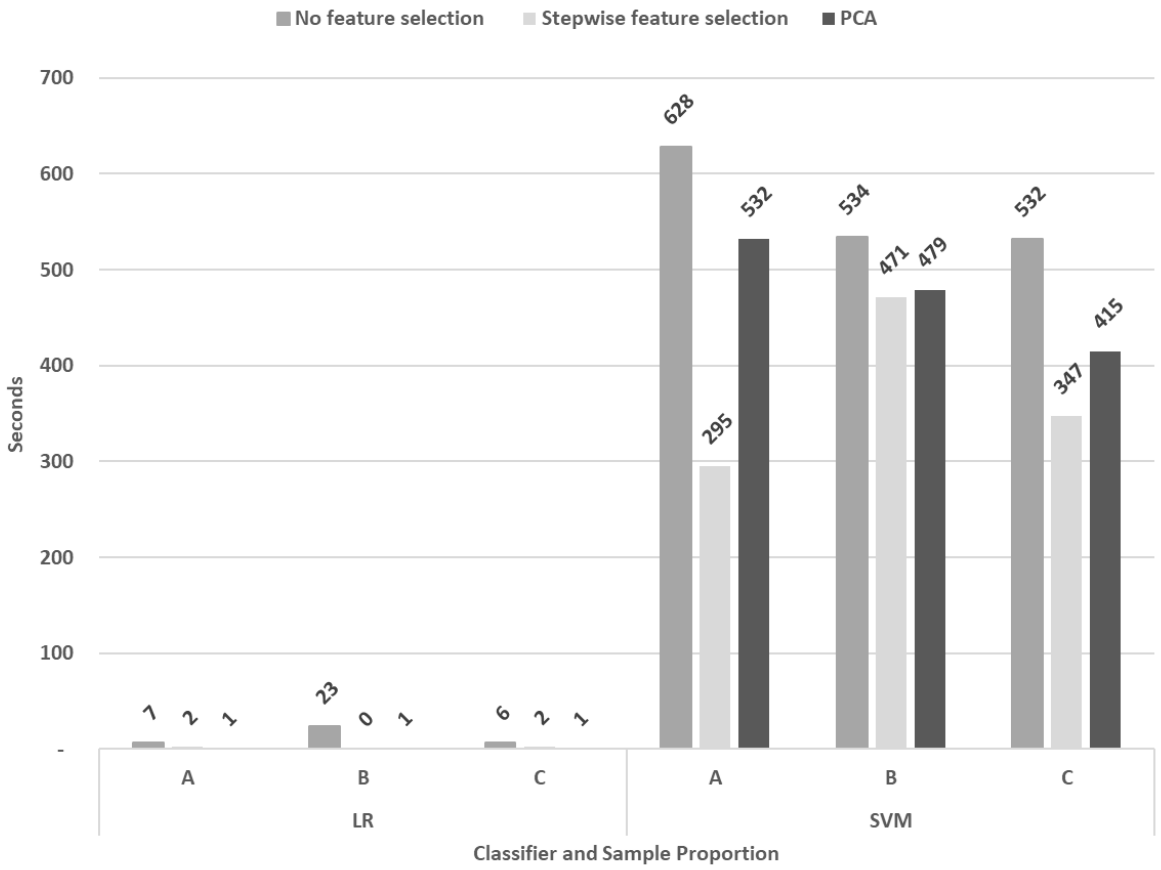

FIGURE 5.9: *Run time summary per classifier, sample proportion and feature selection method.*

applied to the prepared data, followed by analysis of the performance metrics and diagnostics of either model.

Various models performed with good results, with the sensitivity of some models reaching just below 100%. The accuracy levels of some of the models were at 85%, which is very high in a practical, real-life situation. The post application diagnostics showed that the sample size (n=500,000) was too large for the Hosmer-Lemeshow test, causing the test to fail. However, considering the high performance metrics and the fact that the models achieved favourable results in the McFadden's Pseudo R^2 hypothesis test means that a single diagnostic test is inferior to multiple diagnostic tests used in conjunction with one another.

The major classification performance findings were that the sample proportion plays a significant role in the accuracy, sensitivity and specificity; however, it can be seen that feature selection has a limited impact on the classification accuracy of these models on this data set. The feature selection method does have a large impact on the run time of each model. The larger the data set the longer the model runs for, which is why feature selection should be used when the model runs for an extended duration.

CHAPTER 6

Discussion and Conclusions

Contents

This chapter considers the overall discussion of all the findings in the literature review chapter and the results chapter. A summary of the study is then given followed by a comparison between the goals of the study and the results achieved. The contributions made by the study are then stated followed by some possible future areas of research. Recommendations are then discussed after which the ultimate conclusion is stated.

6.1 Discussion

The main discussion points are: the impact of model complexity, the importance of sensitivity in fraud, model conditions and diagnostics, the problem of unbalanced data and the effect of feature selection on model performance.

6.1.1 Model Complexity

SVM is computationally more complex than LR. This means that training the model takes longer when using an SVM model even when the training sample was ten times smaller than the LR sample. This is made apparent in Figure 5.9 where the SVM models run several times longer than the LR models. The data used to train these models is not the full population and

running the SVM on the full population would be highly impractical. This raises the question: "How applicable is SVM to this data set?".

A major strength of SVMs is its ability to perform well with very small data sets. The DO data set is not specifically large; however, it is already too large for SVM models to be trained within a reasonable time span. Considering the run time results for LR it is much more practical to use a simple model like LR with this data set. The run time results for LR are much lower than that of SVM. With approximately the same accuracy, specificity and sensitivity for both models, but a much lower run time for LR, it is clear that LR is more applicable to this data set. When comparing LR and SVM, Kuhn *et al.* [38] find there is no performance loss by using a more straightforward model. The same result is reached in this research. The next discussion point is the importance of sensitivity.

6.1.2 The Importance of Sensitivity

According to Peussa *et al.* [55], in the case of credit default, sensitivity is more important than specificity or accuracy. This is the case for fraud detection as well. The results in Figures 5.6 to 5.8 show that high levels of sensitivity can be achieved by altering the number of fraudulent cases in the sample. There were models that produced sensitivity rates just below 100%; however, this is at the cost of very low specificity rates for the same models.

High sensitivity rates are very desirable when fraud is considered, even at the cost of low specificity rates. There is a point where the specificity rate becomes too low and the false positive rate becomes so high that the cost to investigate all the false positives begins to out-weigh the gains from preventing the fraud, this is the probable scenario when 100% sensitivity is achieved. Sampling proportions are very important to the sensitivity and specificity relationship as it dictates the behaviour of the two measures.

The ratio of sensitivity and specificity needs to be determined on a case-by-case basis to achieve the result that is required for a specific situation. The break-even point for a bank for this trade-off is different to a telecommunications company. To say either specificity or sensitivity is superior without sufficient motivation would imply a misleading result. The next discussion points are the conditions and diagnostics.

6.1.3 Model Conditions and Diagnostics

The LR models all rejected the Hosmer-Lemeshow hypothesis test indicating that there was an underlying issue with the data. This was not the case though, since the sample size was too large to apply the Hosmer-Lemeshow hypothesis test. Most models achieved satisfactory McFadden's Pseudo R^2 values, indicating that the models had high predictive power. Looking at Table 5.4, it is clear that the LR models produced relatively accurate results. The above three points show that it is important to meet specific modelling conditions before applying any classifier to a data set and to perform diagnostics post-application.

Awoyemi *et al.* [1] considers LR and SVM in fraud detection and finds that LR performs at

a lower accuracy than SVM. The results found in this research, using the DO data, indicate that LR and SVM are approximately equivalent. This is supported by the findings of Oza [49], who concludes that both models perform with high accuracy. Awoyemi *et al.* [1] do not indicate whether condition testing was done before models were applied. It is also not indicated whether diagnostic tests were conducted after the models were applied. There may have been an underlying data issues which were not noticed.

The above discussion demonstrates the importance of diagnostic tests. They are not necessarily conclusive as they all have shortcomings, but to exclude any condition or diagnostics tests would be unwise. Without condition and diagnostic tests it becomes possible to undermine a result, even if the result is correct. The next discussion point is the problem of unbalanced data.

6.1.4 The Problem of Unbalanced Data

Unbalanced data is an issue that is prevalent within the fraud environment, refer to Section 3.4. Itoo *et al.* [30] suggested the use of under-sampling to overcome this issue. The same issue was present here and sample proportions chosen to assess its impact were: 50%-50%, 80%-20% and 20%-80% of fraudulent to non-fraudulent observations.

The performance results varied dramatically depending on the sample proportion chosen. The 80%-20% sample proportion produces the highest accuracy for both LR and SVM and the highest sensitivity for both classification methods as well. This came at the cost of very low specificity rates. The 20%-80% sample proportion produces slightly lower accuracy results with the highest specificity rates. This was at the cost of low sensitivity rates and is not a desirable result when fraud classification is considered.

Sample proportion B produces the lowest accuracy rates, but has more balanced sensitivity and specificity rates. The models produced by this sample proportion is the most balanced overall and depending on the situation is highly desirable. The last component to be considered in the Discussion section is the effect of feature selection.

6.1.5 The Effect of Feature Selection

When looking at the feature selection on all models there is very little difference in performance between stepwise feature selection and no feature selection. The only notable difference in performance is where run time is considered where stepwise feature selection models would produce a lower run time than no feature selection models. This result means that the models do not necessarily need feature selection to produce accurate results, but on larger data sets results will be produced faster. This is an important point when real world application of these models is concerned.

Ravisankar *et al.* [60] suggest that feature selection is critical to data mining. This is true in situations where large data sets are concerned and where the run time has a financial impact on a business.

The above sections explore the major discussion points relative to the literature and the findings in Chapter 5. It is found that LR is more applicable to this setting than SVM. The greater relative importance of sensitivity to specificity and accuracy is demonstrated. It is determined that model condition and diagnostic testing is of critical importance. Unbalanced data is discussed, showing that different sampling proportions can produce vastly different model performance results. It is also shown, with this data set, that feature selection makes little difference to model performance, but makes a large difference to model run time. The next section will consider the conclusions based on the above discussion.

6.2 Conclusions

The final section of this study gives a summary of the study along with the goals achieved, followed by contributions and suggestions of future research. Recommendations are given followed by the final component which is the study conclusion.

6.2.1 Study Overview

In summary, with R74million worth of fraudulent DOs per month, the need for fraud detection is clear. The weakness was identified as the placement of the mandate with the collector (seller) as opposed to the banks. The appropriate ML models, (LR and SVM) for this type of fraud, were selected based on the conditions of the data and other research relevant to fraud. The ML models were then applied to the case study data.

Efforts by banks to reduce fraudulent DOs are seen in the decreasing proportion of fraud per week, shown in Figure 4.2. These efforts are also seen when variables such as *R99Flag* are present in the data to assist with the ease of identification. The A/C debit order system is an attempt from the NPS to curb the shortcomings in the four party payment system. The ML fraud detection method complements the above mentioned efforts in the fight against automated payment fraud.

It was found that feature selection had a small impact on the performance of models; however, it was evident that the sample proportions selected had a major impact on the model performance. This was an exercise of fraud identification where sensitivity is the most important. For this reason the 80%-20% fraudulent to non-fraudulent sample proportion is the most applicable to this situation as the highest sensitivity rates were produced. This may not be the case for all data sets and environments as the cost to investigate false positives may be higher than the actual cost of fraud prevented. The sample proportion and associated sensitivity and specificity rates need to be considered on a case-by-case basis.

Condition testing and post model application diagnostics were applied in this research. It was evident PCA feature selection was inferior to stepwise feature selection in this case due to diagnostic failures and relatively poor performance results. The relatively poor performance of the PCA feature selection models is due to a loss of information when variables are removed for having KMO values below 0.7.

When considering the odds ratios for LR, there were several variables that were protective factors and others that were risk factors. These factors either increased or decreased the odds of a case being fraudulent. It was found that when a DO belonged to an older person it was more likely to be fraudulent than when the DO belonged to a younger person. It was also found that if a DO had a value of R99 then the odds of the case being fraudulent would increase several fold. This ties in with the "R99 scam" that was identified in Section 2.7.

LR models produced equivalent results to the more complex SVM models with a much better run time. From a practical point of view, this means that LR should perform better on larger data sets. Only one year worth of data is considered in this study, it would not be practical to apply SVM models to the population data set. For this reason LR is the better classifier for this situation.

Considering the above summary it can be concluded that ML techniques can be used to classify fraudulent DOs with high levels of accuracy.

6.2.2 Objectives Achieved

The objectives specified in Section 1.3 are listed in Table 6.1 with status of whether it was achieved or not.

Section Design	Objective	Status
Literature Review	Describe the components of the payment system, interoperability and the four party model	Achieved
Literature Review	Define debit orders and the different types of payments systems	Achieved
Literature Review	Describe debit order disputes and abuse	Achieved
Literature Review	Describe fraud in the DO environment	Achieved
Literature Review	Identify applicable models to identify fraud	Achieved
Case Study	Describe conditions for ML model application	Achieved
Case Study	Apply ML models to the case study data	Achieved
Case Study	Define and apply model diagnostics and performance measurement so that the results are valid and reliable	Achieved

TABLE 6.1: *List of objectives achieved.*

As can be seen in Table 6.1 all of the objectives laid out at the beginning of the research were achieved. The main research question, specified in Section 1.2, can therefore be restated and answered: "Can fraudulent DOs be identified using the machine learning process, LR and SVMs?". Yes, it can be concluded that ML techniques can be used to identify fraudulent DOs.

The expectation of valid and reliable results was created in Section 1.6. The predicted error in the Hosmer-Lemeshow graphs (see Figures 5.1 to 5.3) is compared against the predicted error rate produced by the testing data set (see Table 5.5) and is found to produce similar error rates, indicating a reliable result. The condition testing, model diagnostics and similar predicted error rates means the results produced are valid and reliable.

6.2.3 Contribution

This is the first time DO fraud has been investigated using ML in South Africa. There is a large amount of credit card fraud research (globally), but this type of automated payment fraud has not yet been researched in South Africa. This research sheds light on an area of study that has not yet been analysed. The critical testing of results is not always present in research and is present in this study. Since conclusions are reached with rigorous condition and diagnostic testing, the methodology discussed in this study can be applied to other studies in fraud. The final contribution is an ML model that can be applied in the banking industry. This model can be applied directly at participating banks, where it can begin identifying fraudulent DOs. This means banks are able to: save money for their clients, add additional intelligent financial technology to their business and increase the quality of their reputation due to less DO fraud. These are significant contributions in both the academic sphere and in the financial technology market.

6.2.4 Future Research

There are several areas where future research can be conducted. The kernel used in this study was the RBF kernel; however, there are many other kernels that could be considered. The use of other kernels on the same data set could produce higher performance results and is a topic that could be researched in future. Feature selection had a minor impact on the performance of models; however, there are other feature selection methods that could be used to see if there is a further run time gain and if there is a larger gain in model performance. The sample proportions chosen in this research is not an exhaustive list of proportion combinations. A future study can be conducted to find the optimal balance between fraudulent and non-fraudulent observations to produce the highest model performance without the large costs in sensitivity and specificity. Since the cut-off point for LR was fixed at 0.5 for the duration of the study, the optimal cut-off point for LR models can also be explored. Only LR and SVM were considered in this study; however, there are many other classification methods that could be explored with the possibility of producing even better performance results. This study considers only fraudulent DOs and there are many studies on credit card fraud detection; however, these methods can be applied in any fraud detection environment such as identity fraud. This is an area where major research can still be conducted.

6.2.5 Recommendations

There is a clear need for intervention where fraud is concerned; however, traditional methods of fraud detection are falling short due to the rapidly changing nature of both fraud and the financial technology market. It is recommended that alternative methods of fraud detection methods are explored. The use of ML is the obvious recommendation due to its ability to interpret ever growing stockpiles of data. The use of computers to sift through the data by means of an algorithm is the logical next step for any enterprise to increase security of their systems. It is to recommend to use ML to identify fraud where ever there is a need for fraud detection.

When it comes to using a ML model to identify fraud, it is recommended that sample proportions of non-fraudulent to fraudulent observations are considered and that feature selection is

used when the data set is large. It is further recommended to use LR when the data set is large and SVM when the number of features is large. In spite of all of the above recommendations it is proposed that the correct algorithm is used for the data set along with the correct feature selection method. Certain models fit certain data sets better than other models and this flexibility is a necessity to achieve the best possible results.

In conclusion, there will always be loopholes in banking systems and there will always be a group of individuals willing to find and exploit these loopholes [64]. The ML approach of fraud identification offers a longer term approach to combat the advances of fraudsters. The consequences of reducing fraud are felt by many individuals in South Africa and has a positive effect on the NPS; therefore, every effort must be made to eradicate the "rogue elements" in the system. Once the current fraud is removed, the NPS must be made more secure. The question then becomes: "what will the new fraud look like?".

List of references

[1] Awoyemi JO, Adetunmbi AO and Oluwadare SA, 2017, *Credit card fraud detection using machine learning techniques: A comparative analysis*, Proceedings of the 2017 International Conference on Computing Networking and Informatics (ICCNI), pp. 1–9.

[2] Bacs, *Direct debit consultation outcomes*, [Online], [Cited 13 March 2019], Available from `https://www.bacs.co.uk/documentlibrary/dd_consultation_outcomes.pdf`.

[3] Banking Enquiry, 2008, *Report to the Competition Commissioner by the Enquiry Panel, Chapter 4*.

[4] Banking Enquiry, 2008, *Report to the Competition Commissioner by the Enquiry Panel, Chapter 7*.

[5] Bell E, Bryman A and Harley B, 2018, *Business research methods*, Oxford university press.

[6] Ben-Hur A and Weston J, 2010, *A user's guide to support vector machines*, pp. 223–239 in *Data mining techniques for the life sciences*, pp. 223–239. Springer.

[7] Bhattacharyya S, Jha S, Tharakunnel K and Westland JC, 2011, *Data mining for credit card fraud: A comparative study*, Decision Support Systems, **50(3)**, pp. 602–613.

[8] Bujang MA, Sa'at N, Bakar TMITA *et al.*, 2018, *Sample size guidelines for logistic regression from observational studies with large population: emphasis on the accuracy between statistics and parameters based on real life clinical data*, The Malaysian Journal of Medical Sciences: MJMS, **25(4)**, p. 122.

[9] Businesstech, *Capitec fights back against R99 debit order fraud*, [Online], [Cited 21 October 2019], Available from `https://businesstech.co.za/news/banking/325417/capitec-fights-back-against-r99-debit-order-fraud/amp/`.

[10] Businesstech, *Organised crime syndicates are using fake debit orders to take billions of rands from South Africans: report*, [Online], [Cited 21 October 2019], Available from `https://businesstech.co.za/news/banking/320915/organised-crime-syndicates-are-using-fake-debit-orders-to-take-billions`.

[11] Camm JD, Cochran JJ, Fry MJ, Ohlmann JW and Anderson DR, 2014, *Essentials of Business Analytics (Book Only)*, Nelson Education.

[12] Chandrashekar G and Sahin F, 2014, *A survey on feature selection methods*, Computers & Electrical Engineering, **40(1)**, pp. 16–28.

[13] Committee on Payments and Market Infrastructure, 2012, *Payment, clearing and settlement systems in South Africa*.

[14] Coppolino L, D'Antonio S, Romano L, Papale G, Sgaglione L and Campanile F, 2015, *Direct debit transactions: a comprehensive analysis of emerging attack patterns*, Proceedings of the 2015 10th International Conference on P2P, Parallel, Grid, Cloud and Internet Computing (3PGCIC), pp. 713–717.

[15] Dey A, 2016, *Machine learning algorithms: a review*, International Journal of Computer Science and Information Technologies, **7(3)**, pp. 1174–1179.

[16] DirectDebit, *Reduce your debit order dispute rates*, [Online], [Cited 13 March 2019], Available from `https://www.debitorder.com/debit-orders/reduce-debit-order-dispute-rates/`.

[17] Dobson AJ and Barnett A, 2008, *An introduction to generalized linear models*, CRC press.

[18] Dodge Y, 2008, *The concise encyclopedia of statistics*, Springer Science & Business Media.

[19] Dreiseitl S and Ohno-Machado L, 2002, *Logistic regression and artificial neural network classification models: a methodology review*, Journal of Biomedical Informatics, **35(5-6)**, pp. 352–359.

[20] Dubey R, Zhou J, Wang Y, Thompson PM, Ye J, Initiative ADN *et al.*, 2014, *Analysis of sampling techniques for imbalanced data: An n= 648 adni study*, NeuroImage, **87**, pp. 220–241.

[21] Dziuban CD and Shirkey EC, 1974, *When is a correlation matrix appropriate for factor analysis? Some decision rules.*, Psychological Bulletin, **81(6)**, p. 358.

[22] Finklea KM, 2009, *Identity theft: Trends and issues*, DIANE Publishing.

[23] Frenzel P, Schroth C and Samsonow T, 2007, *The enterprise interoperability center-an institutional framework facilitating enterprise interoperability.*

[24] Grus J, 2019, *Data science from scratch: first principles with python*, O'Reilly Media.

[25] Gujarati DN and Porter DC, 1999, *Essentials of econometrics*, volume 2, Irwin/McGraw-Hill Singapore.

[26] Harrell Jr FE, Lee KL, Califf RM, Pryor DB and Rosati RA, 1984, *Regression modelling strategies for improved prognostic prediction*, Statistics in Medicine, **3(2)**, pp. 143–152.

[27] Hosmer Jr DW, Lemeshow S and Sturdivant RX, 2013, *Applied logistic regression*, volume 398, John Wiley & Sons.

[28] Hsu CW, Chang CC, Lin CJ *et al.*, 2003, *A practical guide to support vector classification.*

[29] Ismail A, *Banks under fire over R99 debit order scam*, [Online], [Cited 05 November 2019], Available from `https://www.fin24.com/Companies/Financial-Services/banks-under-fire-over-r99-debit-order-scam-20151120`.

[30] Itoo F, Singh S *et al.*, 2020, *Comparison and analysis of logistic regression, Naïve Bayes and KNN machine learning algorithms for credit card fraud detection*, International Journal of Information Technology, pp. 1–9.

[31] James G, Witten D, Hastie T and Tibshirani R, 2013, *An introduction to statistical learning*, volume 112, Springer.

[32] Kaiser HF, 1970, *A second generation little jiffy*, Psychometrika, **35(4)**, pp. 401–415.

[33] Kaiser HF and Rice J, 1974, *Little jiffy, mark IV*, Educational and Psychological Measurement, **34(1)**, pp. 111–117.

[34] Karrim A, Cowan K and Evans S, *Organised crime behind massive R1.6bn debit order scam*, [Online], [Cited 09 July 2019], Available from `https://www.fin24.com/Economy/exclusive-organised-crime-behind-massive-r16bn-debit-order-scam-20190602-2`.

[35] Kassambara A, 2018, *Machine Learning Essentials: Practical Guide in R*, sthda.

[36] Kleine J, Venzin M, Munisso A and Weller T, 2010, *SEPA Direct Debit - a success story for the European payment market*.

[37] Kotsiantis SB, Zaharakis I and Pintelas P, 2007, *Supervised machine learning: A review of classification techniques*, Emerging Artificial Intelligence Applications in Computer Engineering, **160**, pp. 3–24.

[38] Kuhn M, Johnson K *et al.*, 2013, *Applied predictive modeling*, volume 26, Springer.

[39] Louviere JJ, Hensher DA and Swait JD, 2000, *Stated choice methods: analysis and applications*, Cambridge University Press.

[40] Mansfield ER and Helms BP, 1982, *Detecting Multicollinearity*, The American Statistician, **36(3)**, pp. 158–160, Available from `http://www.jstor.org/stable/2683167`.

[41] Mathur A and Foody GM, 2008, *Multiclass and binary SVM classification: Implications for training and classification users*, IEEE Geoscience and Remote Sensing Letters, **5(2)**, pp. 241–245.

[42] Mitra P, Murthy C and Pal SK, 2002, *Unsupervised feature selection using feature similarity*, IEEE Transactions on Pattern Analysis and Machine Intelligence, **24(3)**, pp. 301–312.

[43] MyBroadband, *Huge growth in debit order disputes in South Africa.*, [Online], [Cited 13 March 2019], Available from `http://mybroadband.co.za/news/banking/267757-huge-growth-in-debit-order-disputes-in-south-africa.html`.

[44] National Payment System Department, 2011, *Position Paper on Interoperability*.

[45] Niselow T, *Are authenticated collections reinventing the debit order?*, [Online], [Cited 15 November 2019], Available from `https://www.fin24.com/Tech/Opinion/are-authenticated-collections-reinventing-the-debit-order-20170509`.

[46] Ohsaki M, Wang P, Matsuda K, Katagiri S, Watanabe H and Ralescu A, 2017, *Confusion-matrix-based kernel logistic regression for imbalanced data classification*, IEEE Transactions on Knowledge and Data Engineering, **29(9)**, pp. 1806–1819.

[47] Omarjee L, *Debit order fraud: Beware of sharing your banking details*, [Online], [Cited 29 October 2019], Available from `https://www.fin24.com/Money/Home/debit-order-fraud-beware-of-sharing-your-banking-details-20181106`.

[48] Ombudsman for Banking Services, *Bulletin 11 - debit orders*, [Online], [Cited 25 April 2019], Available from https://www.obssa.co.za/wp-content/uploads/2018/02/Bulletin-11-Debit-Orders-Final-30.01.2018.pdf.

[49] Oza A, 2018, *Fraud Detection using Machine Learning*, TRANSFER, **528812(4097)**, p. 532909.

[50] Paul P, Pennell ML and Lemeshow S, 2013, *Standardizing the power of the Hosmer-Lemeshow goodness of fit test in large data sets*, Statistics in Medicine, **32(1)**, pp. 67–80.

[51] Payments Association of South Africa, *Amendment of the directive for conduct within the NPS in respect of the collection of payemt instructions of authenticated collections*, [Online], [Cited 22 March 2019], Available from https://www.gov.za/sites/default/files/gcis_document/201812/42100gen801.pdf.

[52] Payments Association of South Africa, *Directive (no.2) for conduct within the national payment system*, [Online], [Cited 21 March 2019], Available from http://www.pasa.org.za/docs/default-source/default-document-library/resources/sarb-directive-2-of-2007-for-system-operators-(1).pdf?sfvrsn=2.

[53] Payments Association of South Africa, *What is a debit order*, [Online], [Cited 11 March 2019], Available from http://www.pasa.org.za/resources/debit-order.

[54] Payments Association of South Africa, 2017, *Technical requirements specification authenticated collections*.

[55] Peussa A *et al.*, 2016, *Credit risk scorecard estimation by logistic regression*.

[56] Pillay K, *Debit order headache*, [Online], [Cited 21 October 2019], Available from https://www.news24.com/SouthAfrica/News/debit-order-headache-20181101.

[57] Pochiraju B and Seshadri S, 2018, *Essentials of Business Analytics: An Introduction to the Methodology and Its Applications*, volume 264, Springer.

[58] R Core Team, 2017, *R: A Language and Environment for Statistical Computing*, R Foundation for Statistical Computing, Vienna, Austria, Available from https://www.R-project.org/.

[59] Raschka S, 2015, *Python Machine Learning*, Packt Publishing.

[60] Ravisankar P, Ravi V, Rao GR and Bose I, 2011, *Detection of financial statement fraud and feature selection using data mining techniques*, Decision Support Systems, **50(2)**, pp. 491–500.

[61] Ricci C, Baumgartner J, Wentzel-Viljoen E and Smuts CM, 2019, *Food or nutrient pattern assessment using the principal component analysis applied to food questionnaires. Pitfalls, tips and tricks*, International Journal of Food Sciences and Nutrition, **70(6)**, pp. 738–748.

[62] Ripley BD, 2002, *Modern applied statistics with S*, springer.

[63] Rummel RJ, 1988, *Applied factor analysis*, Northwestern University Press.

[64] Sibindana D, *Banks must act on debit order fraud*, [Online], [Cited 05 November 2019], Available from https://www.fin24.com/Opinion/banks-must-act-on-debit-order-fraud-20151209.

[65] Singh G, Gupta R, Rastogi A, Chandel MD and Ahmad R, 2012, *A machine learning approach for detection of fraud based on svm*, International Journal of Scientific Engineering and Technology, **1(3)**, pp. 192–196.

[66] Smith C, *Significant increase in debit order disputes*, [Online], [Cited 04 November 2019], Available from `https://www.fin24.com/Money/Banking/significant-increase-in-debit-order-disputes-20180712`.

[67] Soanes C and Stevenson A, 2004, *Concise oxford english dictionary*, volume 11, Oxford University Press Oxford.

[68] South African Revenue Services, *National Payment System act 78 of 1998*, [Online], [Cited 04 May 2019], Available from `https://www.resbank.co.za/RegulationAndSupervision/NationalPaymentSystem(NPS)/Legal/Documents/NPS-20Act.pdf`.

[69] Stead K, *Pasa cracks down on debit order fraud*, [Online], [Cited 03 November 2019], Available from `https://www.fin24.com/Money/Banking/pasa-cracks-down-on-debit-order-fraud-20180802`.

[70] Subudhi S and Panigrahi S, 2015, *Quarter-sphere support vector machine for fraud detection in mobile telecommunication networks*, Procedia Computer Science, **48**, pp. 353–359.

[71] Sullivan KM and Luke S, 2007, *Evolving kernels for support vector machine classification*, Proceedings of the 9th annual conference on *Genetic* and *Evolutionary Computation*, pp. 1702–1707.

[72] Vermeulen J, *The other debit order fraud – consumers reversing legitimate payments*, [Online], [Cited 13 March 2019], Available from `https://mybroadband.co.za/news/banking/272447-the-other-debit-order-fraud-consumers-reversing-legitimate-payments.html`.

[73] Visa S, Ramsay B, Ralescu AL and Van Der Knaap E, 2011, *Confusion Matrix-based Feature Selection.*, MAICS, **710**, pp. 120–127.

[74] Volker W, 2013, *Essential Guide to Payments: An Overview of the Services, Regulation and Inner Workings of the South Africa National Payment System*, Veritas Books, Available from `https://books.google.co.za/books?id=kXiAngEACAAJ`.

[75] Wah YB, Rahman HAA, He H and Bulgiba A, 2016, *Handling imbalanced dataset using svm and k-NN approach*, Proceedings of the 1st AIP Conference Proceedings, volume 1750, p. 020023.

[76] Yu WF and Wang N, 2009, *Research on credit card fraud detection model based on distance sum*, Proceedings of the 2009 International Joint Conference on Artificial Intelligence, pp. 353–356.

[77] Zikmund WG, Carr JC and Griffin M, 2013, *Business Research Methods (Book Only)*, Cengage Learning.

APPENDIX A

R Code

All R code used in the production of results in Chapter 5.

A.1 LR Code:

A.1.1 Loading packages

The necessary packages are loaded using the below commands.

```
library(car)
library(plyr)
library(RODBC)
library(odbc)
library(ISLR)
library(caret)
library(e1071)
library(MASS)
library(ROCR)
library(pROC)
library(ResourceSelection)
library(tidyverse)
library(broom)
library(DescTools)
library(generalhoslem)
library(psych)
library(dplyr)
```

A.1.2 Data extraction from server

The data is extracted from SQL using the below commands.

```
con <- RODBC::odbcDriverConnect('Driver=SQL Server Native Client 11.0;
                                 SERVER=;
                                 DATABASE=SDB_BI;
                                 trusted_connection=yes;')
```

```
#-------------------------------------------------------------------------------

#Balanced (50-50)
SQL_Query <- sqlQuery(con,"(select top 250000 *
from sdb_bi.dbo.HT_DO_Extract_grouped
where Group_Classification in (1))
union all
(select top 250000 *
from sdb_bi.dbo.HT_DO_Extract_grouped
where Group_Classification in (2))")

#Balanced (80-20)
SQL_Query <- sqlQuery(con,"(select top 400000 *
from sdb_bi.dbo.HT_DO_Extract_grouped
where Group_Classification in (1))
union all
(select top 100000 *
from sdb_bi.dbo.HT_DO_Extract_grouped
where Group_Classification in (2))")

#Balanced (20-80)
SQL_Query <- sqlQuery(con,"(select top 100000 *
from sdb_bi.dbo.HT_DO_Extract_grouped
where Group_Classification in (1))
union all
(select top 400000 *
from sdb_bi.dbo.HT_DO_Extract_grouped
where Group_Classification in (2))")

SQL_Query_backup <- SQL_Query
library(dplyr)
categorical_data<-select(SQL_Query
                         ,Age_band
                        ,App_Activated
                        ,App_Registered
                        ,Banking_Client
                        ,Banking_Client_New
                        ,Employee
                        ,Credit_Card_Client
                        ,Fixed_Savings_Client
                        ,Flexible_Savings_Client
                        ,Gender_code
                        ,Gov
                        ,Grouping_DebiCheck_Branch_Dispute
                        ,Grouping_Branch_Dispute
                        ,Inflow_Current_Month_Grouped
                        ,Official_Client_Type_Fee
                        ,Official_Client_Type_Loan
                        ,Official_Client_Type_New
```

```
                    ,Province_EC
                    ,Province_FS
                    ,Province_G
                    ,Province_KZN
                    ,Province_L
                    ,Province_M
                    ,Province_NW
                    ,Province_NC
                    ,Province_WC
                    ,R45Flag
                    ,R99Flag
                    ,Reverter
                    ,Reverter_New
                    ,Risk_Group_NLR_Compuscore
                    ,Term_Loan_Client_GoodStanding
                    ,Title_DR
                    ,Title_MADAME
                    ,Title_MISS
                    ,Title_MR
                    ,Title_MRS
                    ,Title_MS
                    ,Title_PROF
                    ,Val_group
                    ,Ave_Inflows_6Mnth_Grouped
                    ,Branch_Visits_12mnth_Grouped
                    ,POS_Current_Month_Grouped
                    ,POS_Value_Grouped
                    ,Ave_DO_6Mnth
                    ,Ave_DO_3Mnth
                    ,DO_Current_Month
                    ,DO_Current_Month_Grouped
                    ,Num_ATM_Withdrawals_Current_Month_Grouped
                    ,ATM_Withdrawal_Amount_Grouped
                    ,Quality_Banking_Client
                    ,Stable_Product_Usage_Highly_Stable
                    ,Stable_Product_Usage_Stable
                    ,Stable_Product_Usage_Unstable
                    ,DO_classification
)
#=============================================================================
```

A.1.3 Multicollinearity removal in R

Multicollinearity is removed using the below commands.

```
mydata <- categorical_data
dim(mydata)
mydata$DO_classification = as.numeric(mydata$DO_classification)
fit=lm(DO_classification ~ ., data=mydata)
```

```
vif(fit)
threshold=5
drop=TRUE
aftervif=data.frame()
while(drop==TRUE) {
  vfit=vif(fit)
  aftervif=rbind.fill(aftervif,as.data.frame(t(vfit)))
  if(max(vfit)>threshold) { fit=
    update(fit,as.formula(paste(".","~",".","-",names(which.max(vfit))))) }
  else { drop=FALSE }}
t_aftervif= as.data.frame(t(aftervif))
vfit_d= as.data.frame(vfit)
print(vfit_d)
data_base<-SQL_Query[,c(names(vfit),"DO_classification")]
#=============================================================================
data<-data_base
smp_size <- floor(0.8 * nrow(data))
set.seed(123)
train_ind <- sample(seq_len(nrow(data)), size = smp_size)
train <- data[train_ind, ]
test <- data[-train_ind, ]
ptm <- proc.time()
```

A.1.4 Model application in R (no feature selection)

The logistic regression model is applied to the data set after multicollinearity is removed using the below commands.

```
glm.fit <- glm(DO_classification ~ . , data = train
               , family=binomial(link="logit"))
Run_time<-proc.time() - ptm
glm.fit
library(caret)
pdata <- predict(glm.fit, newdata = test, type = "response")
data = as.numeric(pdata>0.5)
```

A.1.5 Performance measure calculations in R

The following commands are used to produce the model performance measures.

```
confusionMatrix(factor(data), factor(test$DO_classification))
print('Run Time:')
Run_time
hoslem.test((data), (test$DO_classification), g=10)
PseudoR2(glm.fit)
pROC_obj <- roc(data,test$DO_classification,
                smoothed = TRUE,
```

```
                ci=TRUE, ci.alpha=0.7, stratified=FALSE,
                plot=TRUE, auc.polygon=TRUE, max.auc.polygon=TRUE, grid=TRUE,
                print.auc=TRUE, show.thres=TRUE)
sens.ci <- ci.se(pROC_obj)
plot(sens.ci, type="shape", col="lightblue", main = 'LOG')
plot(sens.ci, type="bars")
#=============================================================================
```

A.1.6 Stepwise feature selection in R

Stepwise feature selection is performed using the below commands.

```
data<-data_base
smp_size <- floor(0.8 * nrow(data))
set.seed(123)
train_ind <- sample(seq_len(nrow(data)), size = smp_size)
train <- data[train_ind, ]
test <- data[-train_ind, ]
glm.fit <- glm(DO_classification ~ . , data = train, family = binomial)
StepWise <- stepAIC(glm.fit,direction = "both",trace = FALSE)
StepWise$anova
summary(StepWise)
#=============================================================================
data<-data_base
smp_size <- floor(0.8 * nrow(data))
set.seed(123)
train_ind <- sample(seq_len(nrow(data)), size = smp_size)
train <- data[train_ind, ]
test <- data[-train_ind, ]
ptm <- proc.time()
```

A.1.7 Model application in R (stepwise feature selection)

The logistic regression model is applied to the data set after stepwise feature selection is performed using the below commands.

```
glm.fit.step <- glm(StepWise$formula, data = train
                    , family=binomial(link="logit"))
Run_time<-proc.time() - ptm
summary(glm.fit.step)
#=============================================================================
glm.fit.step
library(caret)
pdata <- predict(glm.fit.step, newdata = test, type = "response")
data = as.numeric(pdata>0.5)
confusionMatrix(factor(data), factor(test$DO_classification))
print('Run Time:')
```

```
Run_time
hoslem.test(train$DO_classification,glm.fit.step$fitted.values , g=11)
PseudoR2(glm.fit.step)
pROC_obj <- roc(data,test$DO_classification,
                smoothed = TRUE,
                ci=TRUE, ci.alpha=0.7, stratified=FALSE,
                plot=TRUE, auc.polygon=TRUE, max.auc.polygon=TRUE, grid=TRUE,
                print.auc=TRUE, show.thres=TRUE)
sens.ci <- ci.se(pROC_obj)
plot(sens.ci, type="shape", col="lightblue", main = 'Step LOG')
plot(sens.ci, type="bars")
#=============================================================================
```

A.1.8 PCA feature selection in R

PCA feature selection is performed using the below commands.

```
data<-data.frame(data_base)
KMO_test <- KMO(data)
KMO_test
#=============================================================
data<-data_base[c(
  "Age_band"
  ,"Banking_Client"
  ,"Credit_Card_Client"
  ,"Flexible_Savings_Client"
  ,"Gov"
  ,"Inflow_Current_Month_Grouped"
  ,"Reverter"
  ,"Ave_Inflows_6Mnth_Grouped"
  ,"Branch_Visits_12mnth_Grouped"
  ,"POS_Current_Month_Grouped"
  ,"Ave_DO_6Mnth"
  ,"DO_Current_Month_Grouped"
  ,"Quality_Banking_Client"
  ,"DO_classification")]
smp_size <- floor(0.8 * nrow(data))
set.seed(123)
train_ind <- sample(seq_len(nrow(data)), size = smp_size)
train <- data[train_ind, ]
test <- data[-train_ind, ]
train_scale <- scale(train[,c(          "Age_band"
                                       ,"Banking_Client"
                                       ,"Credit_Card_Client"
                                       ,"Flexible_Savings_Client"
                                       ,"Gov"
                                       ,"Inflow_Current_Month_Grouped"
                                       ,"Reverter"
                                       ,"Ave_Inflows_6Mnth_Grouped"
```

```
                                                ,"Branch_Visits_12mnth_Grouped"
                                                ,"POS_Current_Month_Grouped"
                                                ,"Ave_DO_6Mnth"
                                                ,"DO_Current_Month_Grouped"
                                                ,"Quality_Banking_Client"
                                                 )])
test_scale <- scale(test[,c(   "Age_band"
                              ,"Banking_Client"
                              ,"Credit_Card_Client"
                              ,"Flexible_Savings_Client"
                              ,"Gov"
                              ,"Inflow_Current_Month_Grouped"
                              ,"Reverter"
                              ,"Ave_Inflows_6Mnth_Grouped"
                              ,"Branch_Visits_12mnth_Grouped"
                              ,"POS_Current_Month_Grouped"
                              ,"Ave_DO_6Mnth"
                              ,"DO_Current_Month_Grouped"
                              ,"Quality_Banking_Client"    )])
#=============================================================
training_set_pca <- princomp(scale(train_scale[,c("Age_band"
                                                  ,"Banking_Client"
                                                  ,"Credit_Card_Client"
                                                  ,"Flexible_Savings_Client"
                                                  ,"Gov"
                                                  ,"Inflow_Current_Month_Grouped"
                                                  ,"Reverter"
                                                  ,"Ave_Inflows_6Mnth_Grouped"
                                                  ,"Branch_Visits_12mnth_Grouped"
                                                  ,"POS_Current_Month_Grouped"
                                                  ,"Ave_DO_6Mnth"
                                                  ,"DO_Current_Month_Grouped"
                                                  ,"Quality_Banking_Client"
)]))
summary(training_set_pca)
training_set_pca_data<-data.frame(training_set_pca$scores)
training_set_pca_data$DO_classification =
  unlist(data.frame(train$DO_classification))
training_set_pca_data<-cbind(training_set_pca_data[,1:14]
                             ,DO_classification=train$DO_classification)
ptm <- proc.time()
```

A.1.9 Model application in R (PCA feature selection)

The logistic regression model is applied to the data set after PCA feature selection is performed using the below commands.

```
pca.fit = glm(formula = DO_classification ~. ,
              family=binomial(link="logit"),
```

```
                data = data.frame(training_set_pca_data))
Run_time<-proc.time() - ptm
test_set_pca <- princomp(scale(test_scale[,c(   "Age_band"
                                                ,"Banking_Client"
                                                ,"Credit_Card_Client"
                                                ,"Flexible_Savings_Client"
                                                ,"Gov"
                                                ,"Inflow_Current_Month_Grouped"
                                                ,"Reverter"
                                                ,"Ave_Inflows_6Mnth_Grouped"
                                                ,"Branch_Visits_12mnth_Grouped"
                                                ,"POS_Current_Month_Grouped"
                                                ,"Ave_DO_6Mnth"
                                                ,"DO_Current_Month_Grouped"
                                                ,"Quality_Banking_Client")]))
test_set_pca_data<-data.frame(test_set_pca$scores)
test_set_pca_data$DO_classification = unlist(test$DO_classification)
prob_pred_pca = predict(pca.fit
                        ,newdata = test_set_pca_data[,1:13], type = 'response')
y_pred = ifelse(prob_pred_pca > 0.5, 1, 0)
#=============================================================
summary(pca.fit)
library(caret)
confusionMatrix(factor(y_pred), factor(test$DO_classification))
print('Run Time:')
Run_time
hoslem.test(training_set_pca_data$DO_classification
            ,pca.fit$fitted.values , g=11)
PseudoR2(pca.fit)
pROC_obj <- roc(y_pred,test$DO_classification,
                smoothed = TRUE,
                ci=TRUE, ci.alpha=0.7, stratified=FALSE,
                plot=TRUE, auc.polygon=TRUE, max.auc.polygon=TRUE, grid=TRUE,
                print.auc=TRUE, show.thres=TRUE)
sens.ci <- ci.se(pROC_obj)
plot(sens.ci, type="shape", col="lightblue", main = 'LOG')
plot(sens.ci, type="bars")
```

A.2 SVM Code:

The SVM code follows a similar layout an process as the LR code.

```
library(car)
library(plyr)
library('RODBC')
library('odbc')
library(ISLR)
library('caret')
```

```
library(e1071)
library(MASS)
library(ROCR)
library(pROC)

con <- RODBC::odbcDriverConnect('Driver=SQL Server Native Client 11.0;
                                 SERVER=;
                                 DATABASE=SDB_BI;
                                 trusted_connection=yes;')
#-------------------------------------------------------------------------------
#Balanced (50-50)
SQL_Query <- sqlQuery(con,"(select top 25000 *
from sdb_bi.dbo.HT_DO_Extract_grouped
where Group_Classification in (1))
union all
(select top 25000 *
from sdb_bi.dbo.HT_DO_Extract_grouped
where Group_Classification in (2))")

#Balanced (80-20)
SQL_Query <- sqlQuery(con,"(select top 40000 *
from sdb_bi.dbo.HT_DO_Extract_grouped
where Group_Classification in (1))
union all
(select top 10000 *
from sdb_bi.dbo.HT_DO_Extract_grouped
where Group_Classification in (2))")

#Balanced (20-80)
SQL_Query <- sqlQuery(con,"(select top 10000 *
from sdb_bi.dbo.HT_DO_Extract_grouped
where Group_Classification in (1))
union all
(select top 40000 *
from sdb_bi.dbo.HT_DO_Extract_grouped
where Group_Classification in (2))")
library(dplyr)
categorical_data<-select(SQL_Query
                         ,Age_band
                         ,App_Activated
                         ,App_Registered
                         ,Banking_Client
                         ,Banking_Client_New
                         ,Employee
                         ,Credit_Card_Client
                         ,Fixed_Savings_Client
                         ,Flexible_Savings_Client
                         ,Gender_code
                         ,Gov
                         ,Grouping_DebiCheck_Branch_Dispute
```

```
                                ,Grouping_Branch_Dispute
                                ,Inflow_Current_Month_Grouped
                                ,Official_Client_Type_Fee
                                ,Official_Client_Type_Loan
                                ,Official_Client_Type_New
                                ,Province_EC
                                ,Province_FS
                                ,Province_G
                                ,Province_KZN
                                ,Province_L
                                ,Province_M
                                ,Province_NW
                                ,Province_NC
                                ,Province_WC
                                ,R45Flag
                                ,R99Flag
                                ,Reverter
                                ,Reverter_New
                                ,Risk_Group_NLR_Compuscore
                                ,Term_Loan_Client_GoodStanding
                                ,Title_DR
                                ,Title_MADAME
                                ,Title_MISS
                                ,Title_MR
                                ,Title_MRS
                                ,Title_MS
                                ,Title_PROF
                                ,Val_group
                                ,Ave_Inflows_6Mnth_Grouped
                                ,Branch_Visits_12mnth_Grouped
                                ,POS_Current_Month_Grouped
                                ,POS_Value_Grouped
                                ,Ave_DO_6Mnth
                                ,Ave_DO_3Mnth
                                ,DO_Current_Month
                                ,DO_Current_Month_Grouped
                                ,Num_ATM_Withdrawals_Current_Month_Grouped
                                ,ATM_Withdrawal_Amount_Grouped
                                ,Quality_Banking_Client
                                ,Stable_Product_Usage_Highly_Stable
                                ,Stable_Product_Usage_Stable
                                ,Stable_Product_Usage_Unstable
                                ,DO_classification      )
data_base<-scale(categorical_data)
#=============================================================================
mydata <- scale(categorical_data)
dim(mydata)
mydata$DO_classification = as.numeric(mydata$DO_classification)
fit=lm(DO_classification ~ ., data=mydata)
vif(fit)
```

```
threshold=5
drop=TRUE
aftervif=data.frame()
while(drop==TRUE) {
  vfit=vif(fit)
  aftervif=rbind.fill(aftervif,as.data.frame(t(vfit)))
  if(max(vfit)>threshold) { fit=
    update(fit,as.formula(paste(".","~",".","-",names(which.max(vfit))))) }
  else { drop=FALSE }}
t_aftervif= as.data.frame(t(aftervif))
vfit_d= as.data.frame(vfit)
data_base<-SQL_Query[,c(names(vfit),"DO_classification")]
#=============================================================================
data<-data_base
smp_size <- floor(0.8 * nrow(data))
set.seed(123)
train_ind <- sample(seq_len(nrow(data)), size = smp_size)
train <- data[train_ind, ]
test <- data[-train_ind, ]
ptm <- proc.time()
svm.fit = svm(formula = DO_classification ~ .,
              data = train,
              type = 'C-classification',
              kernel = 'radial')
Run_time<-proc.time() - ptm
summray(svm.fit)
library(caret)
pdata <- predict(svm.fit, newdata = test, type = "response")
confusionMatrix(factor(pdata), factor(test$DO_classification))
print('Run Time:')
Run_time
pROC_obj <- roc(pdata,test$DO_classification,
                smoothed = TRUE,
                ci=TRUE, ci.alpha=0.9, stratified=FALSE,
                plot=TRUE, auc.polygon=TRUE, max.auc.polygon=TRUE, grid=TRUE,
                print.auc=TRUE, show.thres=TRUE)
sens.ci <- ci.se(pROC_obj)
plot(sens.ci, type="shape", col="lightblue")
plot(sens.ci, type="bars")
#=============================================================================
data<-data_base
smp_size <- floor(0.8 * nrow(data))
set.seed(123)
train_ind <- sample(seq_len(nrow(data)), size = smp_size)
train <- data[train_ind, ]
test <- data[-train_ind, ]
glm.fit <- glm(DO_classification ~ . , data = train, family = binomial)
StepWise <- stepAIC(glm.fit,direction = "forward",trace = FALSE)
StepWise$anova
summary(StepWise)
```

```
#=============================================================================
data<-data_base
smp_size <- floor(0.8 * nrow(data))
set.seed(123)
train_ind <- sample(seq_len(nrow(data)), size = smp_size)
train <- data[train_ind, ]
test <- data[-train_ind, ]
ptm <- proc.time()
svm.fit = svm(formula = StepWise$formula,
              data = train,
              type = 'C-classification',
              kernel = 'radial')
Run_time<-proc.time() - ptm
svm.fit
library(caret)
pdata <- predict(svm.fit, newdata = test, type = "response")
confusionMatrix(factor(pdata), factor(test$DO_classification))
print('Run Time:')
Run_time
pROC_obj <- roc(pdata,test$DO_classification,
                smoothed = TRUE,
                ci=TRUE, ci.alpha=0.8, stratified=FALSE,
                plot=TRUE, auc.polygon=TRUE, max.auc.polygon=TRUE, grid=TRUE,
                print.auc=TRUE, show.thres=TRUE)
sens.ci <- ci.se(pROC_obj)
plot(sens.ci, type="shape", col="lightblue", main = 'Step SVM')
plot(sens.ci, type="bars")
#=============================================================================
data<-data.frame(data_base)
KMO_test <- KMO(data)
KMO_test
#=============================================================
data<-data_base[c( "Age_band"
                  ,"Banking_Client"
                  ,"Credit_Card_Client"
                  ,"Flexible_Savings_Client"
                  ,"Gov"
                  ,"Inflow_Current_Month_Grouped"
                  ,"Reverter"
                  ,"Ave_Inflows_6Mnth_Grouped"
                  ,"Branch_Visits_12mnth_Grouped"
                  ,"POS_Current_Month_Grouped"
                  ,"Ave_DO_6Mnth"
                  ,"DO_Current_Month_Grouped"
                  ,"Quality_Banking_Client"
                  ,"DO_classification")]
smp_size <- floor(0.8 * nrow(data))
set.seed(123)
train_ind <- sample(seq_len(nrow(data)), size = smp_size)
train <- data[train_ind, ]
```

```
test <- data[-train_ind, ]
train_scale <- train[,c( "Age_band"
                        ,"Banking_Client"
                        ,"Credit_Card_Client"
                        ,"Flexible_Savings_Client"
                        ,"Gov"
                        ,"Inflow_Current_Month_Grouped"
                        ,"Reverter"
                        ,"Ave_Inflows_6Mnth_Grouped"
                        ,"Branch_Visits_12mnth_Grouped"
                        ,"POS_Current_Month_Grouped"
                        ,"Ave_DO_6Mnth"
                        ,"DO_Current_Month_Grouped"
                        ,"Quality_Banking_Client"
                         )]
test_scale <- test[,c( "Age_band"
                        ,"Banking_Client"
                        ,"Credit_Card_Client"
                        ,"Flexible_Savings_Client"
                        ,"Gov"
                        ,"Inflow_Current_Month_Grouped"
                        ,"Reverter"
                        ,"Ave_Inflows_6Mnth_Grouped"
                        ,"Branch_Visits_12mnth_Grouped"
                        ,"POS_Current_Month_Grouped"
                        ,"Ave_DO_6Mnth"
                        ,"DO_Current_Month_Grouped"
                        ,"Quality_Banking_Client"
                         )]
#=============================================================
training_set_pca <- princomp(train_scale[,c( "Age_band"
                                            ,"Banking_Client"
                                            ,"Credit_Card_Client"
                                            ,"Flexible_Savings_Client"
                                            ,"Gov"
                                            ,"Inflow_Current_Month_Grouped"
                                            ,"Reverter"
                                            ,"Ave_Inflows_6Mnth_Grouped"
                                            ,"Branch_Visits_12mnth_Grouped"
                                            ,"POS_Current_Month_Grouped"
                                            ,"Ave_DO_6Mnth"
                                            ,"DO_Current_Month_Grouped"
                                            ,"Quality_Banking_Client"
                                             ])
summary(training_set_pca)
training_set_pca_data<-data.frame(training_set_pca$scores)
training_set_pca_data$DO_classification =
  unlist(data.frame(train$DO_classification))
training_set_pca_data<-cbind(training_set_pca_data[,1:13]
                            ,DO_classification=train$DO_classification)
```

```
ptm <- proc.time()
pca.fit <- svm(formula = DO_classification ~. ,
               data = data.frame(training_set_pca_data),
               type = 'C-classification',
               kernel = 'radial'
                ))
Run_time<-proc.time() - ptm
summary(pca.fit)
test_set_pca <- princomp(test_scale[,c( "Age_band"
                                        ,"Banking_Client"
                                        ,"Credit_Card_Client"
                                        ,"Flexible_Savings_Client"
                                        ,"Gov"
                                        ,"Inflow_Current_Month_Grouped"
                                        ,"Reverter"
                                        ,"Ave_Inflows_6Mnth_Grouped"
                                        ,"Branch_Visits_12mnth_Grouped"
                                        ,"POS_Current_Month_Grouped"
                                        ,"Ave_DO_6Mnth"
                                        ,"DO_Current_Month_Grouped"
                                        ,"Quality_Banking_Client"
                                         )])
test_set_pca_data<-data.frame(test_set_pca$scores)
test_set_pca_data$DO_classification = unlist(test$DO_classification)
prob_pred_pca = predict(pca.fit
                        ,  newdata = test_set_pca_data[,1:13]
                        , type = 'response')
#=============================================================
library(caret)
confusionMatrix(prob_pred_pca, factor(test$DO_classification))
print('Run Time:')
Run_time
pROC_obj <- roc(prob_pred_pca ,test$DO_classification,
                smoothed = TRUE,
                ci=TRUE, ci.alpha=0.7, stratified=FALSE,
                plot=TRUE, auc.polygon=TRUE, max.auc.polygon=TRUE, grid=TRUE,
                print.auc=TRUE, show.thres=TRUE)
sens.ci <- ci.se(pROC_obj)
plot(sens.ci, type="shape", col="lightblue", main = 'LOG')
plot(sens.ci, type="bars")
```